Confluence results for the pure strong categorical logic C.C.L. -calculi as sub-systems of C.C.L.

Thérèse Hardin

UNITÉ DE RECHERCHE
INRIA-ROCQUENCOURT

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél.:(1) 39 63 55 11

Rapports de Recherche

N° 822

CONFLUENCE RESULTS FOR THE PURE STRONG CATEGORICAL LOGIC C.C.L. λ- CALCULI AS SUB-SYSTEMS OF C.C.L.

Thérèse HARDIN

AVRIL 1988

RR 822

Confluence Results for The Pure Strong Categorical Logic C.C.L.
λ-calculi as sub-systems of C.C.L.

Résultats de confluence pour la Logique Catégorique Forte
Liens avec les λ-calculs

Thérèse Hardin

I.N.R.I.A. (Projet FORMEL) [1]
and Université de Reims[2]

[1]B.P. 105 78150 Le Chesnay CEDEX France
[2]B.P. 347 51062 Reims CEDEX France

Abstract :

The Strong Categorical Combinatory Logic, (C.C.L. , $CCL\beta\eta SP$), developed by P.L. Curien [6] is, when typed and augmented with a rule defining a terminal object, a presentation of Cartesian Closed Categories. Furthermore, it is equationnaly equivalent to the λ-calculus with explicit couples and Surjective Pairing. Here we study the confluence properties of (C.C.L. , $CCL\beta\eta SP$) and of several of its subsystems and the relationship between untyped Lambda-calculi and (C.C.L. , $CCL\beta\eta SP$) as rewriting systems. We prove that there exists a subset $\mathcal{D}$ of C.C.L. and a sub-system $SL\beta\eta$ of $CCL\beta\eta SP$ confluent on $\mathcal{D}$, a very simple isomorphism between Λ, the classical λ-calculus, and a subset $\mathcal{SD}_\lambda$ of $\mathcal{D}$, which is extended between $\beta\eta$-derivations of Λ and a class of derivations of $SL\beta\eta$. Substitution, which is a one step operation belonging to the meta-language of Λ, is now described by rewritings with $SL\beta\eta$ and calculations between several substitutions launched at the same time may be performed by $SL\beta\eta$. This point is a real increasing of the calculation capacities of the Lambda- calculus.

The same result holds for the Lambda-calculus with couples and projection rules (without Surjective Pairing).

The locally confluent sub-system $CCL\beta SP$ (that is $SL\beta\eta+$ (SP)) is not confluent. This result is obtained by firstly designing a new counter-example (different from J.W. Klop's one) for confluence of the lambda-calculus with couples and Surjective Pairing and then translating it into C.C.L. However $CCL\beta\eta SP$ is shown to be confluent on the set derived from $\mathcal{SD}_\lambda$.

We cannot obtain these results with classical methods of confluence and we designed a new method called Intrepretation Method.

Résumé :

La Logique Catégorique Forte, (C.C.L.,$CCL\beta\eta SP$), introduite par P.L. Curien[6], est , une fois typée et augmentée d'une règle pour l'objet terminal, une présentation des Catégories Cartésiennes Fermées. Elle est aussi une théorie équivalente au λ-calcul avec couples explicites et Surjective Pairing. Nous étudions les propriétés de confluence du système de réécriture associé ainsi que ses liens avec les λ-calculs. Nous montrons qu'il existe un sous-ensemble de termes $\mathcal{D}$ et un sous-système $SL\beta\eta$ confluent sur $\mathcal{D}$, un isomorphisme entre Λ (i.e. le λ-calcul classique) et un sous-ensemble $\mathcal{SD}_\lambda$ de $\mathcal{D}$ qui se prolonge en un isomorphisme entre les $\beta\eta$-dérivations de Λ et une classe de dérivations de $SL\beta\eta$ tels que la substitution, opération décrite dans le méta-langage de Λ, soit fragmentée en étapes de réécriture par $SL\beta\eta$. Il est dès lors possible d'effectuer des calculs entre plusieurs substitutions en cours d'évaluation. Ces résultats s'étendent au λ-calcul avec couples (sans (SP)).

Nous montrons que $CCL\beta SP$ ($SL\beta\eta+$ (SP)), bien que localement confluent, n'est pas confluent, en bâtissant un contre-exemple à la confluence de λ-calcul avec couples différent de celui de J.W. Klop. Cependant, restreint aux dérivés de$\mathcal{SD}_\lambda$, le système $CCL\beta\eta SP$ complet est confluent.

L'obtention de ces résultats nécessite une nouvelle méthode appelée Méthode d'interprétation.

Introduction and Summary

The theories of classical Combinators [7,8] and λ-calculus [5,1] both have the same purpose: to describe some of the most primitive and general properties of operations and of their combinations. They are in fact abstract programming languages: higher-order level functional languages can be translated in these theories in order to study strategies, parameter passing problems ... The pure λ-calculus, $\Lambda(\mathrm{V})$, is a formal system built with a set V of variables and two operations: the abstraction of a variable in a term which constructs a new function "of this variable" and the application which applies a function to an argument. The meaning of its operational rule, called β-reduction, is: the value of a function applied to an effective parameter is obtained by replacing, in the body of this function, all the occurrences of the formal parameter by copies of this effective argument.

This substitution of variables by terms, which is only described in the meta-language, is not straightforward: it has to avoid variable name conflicts. This is the main problem in λ-calculus implementations.

$\Lambda(\mathrm{V})$ is also endowed with one another rule: the η-rule which says, roughly speaking, that two functions which have the same value for any argument, are equal.

CL, the Combinatory Logic, is a way of doing computations without using bound variables [18]. Functions are built up from some primitive ones (classically the combinators S and K) and the application operation. Therefore complications of the λ-calculus' substitution are avoided but the intuitive clarity of λ-notation is completely lost. Furthermore, CL-calculus is weaker than λ-calculus: if their bodies are equal then functions are equal in $\Lambda(\mathrm{V})$, not in CL.

So how can we keep this intuitive simplicity of λ-calculus and avoid these name clashes ? Here are two ways. One may use de Bruijn's notation for λ-terms which replaces bound variables names by integers pointing out the abstractor of the variable. This calculus is denoted by Λ. The substitution operation still belongs to the meta-language. It has to do calculations on de Bruijn's numbers to operate exact reallocations when passing an effective argument.

We are concerned with the second way: the Strong Categorical Combinatory Logic, C.C.L. It is a first-order theory developed by P.L. Curien [6]. Its presentation, named *CCLβηSP*, is directly coming from Cartesian Closed Categories's definition. There are several different ties between C.C.L. and λ-calculus we shall now explain.

Firstly, let $\Lambda_{\mathbf{c,f}}(\mathrm{V})$ be obtained by adding a coupling operator of arity 2 and two projection operators to $\Lambda(\mathrm{V})$. The theory *βηSP* is obtained by adding to the theory *βη* two projection rules and the so-called Surjective Pairing rule (SP), which states that every term is a pair. Now in [6], among other results, P.L. Curien proved that $\Lambda_{\mathbf{c,f}}(\mathrm{V})$ and C.C.L. are two equivalent theories: there exists translations between $\Lambda_{\mathbf{c,f}}(\mathrm{V})$ and C.C.L. such that the translations of two terms of $\Lambda_{\mathbf{c,f}}(\mathrm{V})$, equal in the theory *βηSP*, are equal in the theory *CCLβηSP* and conversely.

Our work is a rather complete study of the rewriting system, also named *CCLβηSP*, obtained by orienting the equations of *CCLβηSP* from left to right. Why focusing on this system ?

1. By adding types and a rule for the terminal object, we obtain a rewriting system for Cartesian Closed Categories [15].

2. There is a straightforward translation of Λ (λ-calculus with de Bruijn notation) into C.C.L. such that the structure of λ-terms is kept. The β-reduction is simulated by a derivation with the subsystem called $\mathcal{CCL\beta SP}$. $\mathcal{CCL\beta SP}$ is locally confluent. But $\Lambda_{\mathbf{c,f}}$ is not confluent (Klop's counter-example [12]). Is this sub-system $\mathcal{CCL\beta SP}$ confluent ?
3. (Λ, $\beta\eta$) is a confluent theory. Is there a confluent sub-system of $\mathcal{CCL\beta\eta SP}$ which reproduces $\beta\eta$-derivations ?

Here are our results:

1. λ-calculus substitution is a one-step operation belonging to the meta-language. It is reproduced in C.C.L. by a derivation with a sub-system of $\mathcal{CCL\beta\eta SP}$ called *Subst* therefore broken in several steps described inside C.C.L. *Subst* is locally confluent and terminating. The proof of termination, obtained jointly with A. Laville [10], could not be obtained with classical termination orderings.
2. There exists a subset $\mathcal{D}$ of C.C.L. and a sub-system $SL\beta\eta$ of $\mathcal{CCL\beta\eta SP}$ such that:
   (a) $SL\beta\eta$ is confluent on $\mathcal{D}$.
   (b) Λ is isomorphic to a subset $\mathcal{SD}_\lambda$ of $\mathcal{D}$. This isomorphism is extended between $\beta\eta$-derivations of Λ and a class of derivations of $SL\beta\eta$. Furthermore, let $\mathcal{D}_\lambda$ be the set of $\mathcal{CCL\beta\eta SP}$-derived terms from $\mathcal{SD}_\lambda$. Then $\mathcal{D}_\lambda$ is a subset of $\mathcal{D}$ and $\mathcal{CCL\beta\eta SP}$ is confluent on $\mathcal{D}_\lambda$. So not only we can compute $\beta\eta$-derivations in C.C.L. and divide the substitution operation in smaller steps but we can also perform calculations between several substitutions being evaluated.
   (c) $\Lambda_{\mathbf{c,f}}$ is isomorphic to a subset $\mathcal{SD}_{P\lambda}$ of $\mathcal{D}$ and this isomorphism may be extended between $\beta\eta P$ derivations of $\Lambda_{\mathbf{c,f}}$ and a class of derivations of $SL\beta\eta$.

   Summarizing, ($\mathcal{D}$, $SL\beta\eta$) is a confluent conservative extension of (Λ, $\beta\eta$) and of ($\Lambda_{\mathbf{c,f}}$, $\beta\eta P$).
3. The locally confluent sub-system $\mathcal{CCL\beta SP}$ is not confluent. This result is obtained by first designing a new counter-example for the confluence of $\beta\eta SP$ and then translating it into C.C.L.

These results about confluence cannot be obtained with classical methods. We designed a general method, called Interpretation Method, based on this trick: a given relation R is confluent on a set X if and only if a relation $\mathcal{E}(R)$ induced by R on a set $\mathcal{E}(X)$ is confluent. So it suffices to find a good notion of interpretation for a given set of terms to obtain confluence (or non-confluence) properties on this set.

1 Preliminaries

To fix our terminology and notations, we collect in this preliminary section some well-known notions and results about them. In the last part of this section, we will present the Categorical Logic and several already known results about it.

1.1 Rewriting Systems

Firstly we recall some well-known results about relations on sets.

Notations

Let R be an internal relation on a set E, called ***reduction***. Let M and $N \in E$. $(M,N) \in R$ is denoted by $(M\ R\ N)$. R^ϵ, R^+, $R^\star$, are respectively the reflexive closure, the transitive closure, the reflexive and transitive closure of R. $=_R$ is the equality defined by R. $R^\star$ is also called the *derivation* relation of R.

A normal form for the relation R is called a *R-normal form*. If it exists and is unique, the R-normal form of a term M is denoted by $R(M)$.

Definition 1.1 1. A reduction R is *weakly confluent* if:

$$\forall M \in E\,,\ (M\ R\ N) \text{ and } (M\ R\, P)\ \Rightarrow \exists\, Q \in E\,.\,(N\ R^\star\ Q) \text{ and } (P\ R^\star\ Q)$$

2. A reduction R is *strongly confluent* if:

$$\forall M \in E\,,\ (M\ R\ N) \text{ and } (M\ R\ P) \Rightarrow \exists Q \in E\,.\,(N\ R\ Q) \text{ and } (P\ R\ Q)$$

3. A reduction R is *confluent* if $R^\star$ is weakly (or strongly !) confluent.
4. A reduction R is *confluent* on the term $A \in E$ if the restriction of R to $\{M|M = R^\star(A)\}$ is confluent.
5. A relation is *noetherian* (or is terminating) if $R^\star$ is well-founded, so if there exists no infinite sequence $M_1\ R\ M_2 \ldots\ R\ M_n\ \ldots$.

Extensions of relations

For the formalisation of our results, we need to slightly extend the well-known notion of conservative extension of a given relation.

Definition 1.2 Let R be a relation on a set A and S be a relation on a set B. Suppose that:

1. There exists an injection ϕ from A into B (in the classical definition, ϕ is the inclusion of sets).
2. $\forall M,\, N \in A,\ \ (\phi(M)\, S\, \phi(N)) \ \Leftrightarrow\ (M\, R\, N)$
3. $\forall M,\, N \in A$, if $(\phi(M)\ S\ P)$, then there exists $N \in A$ such that $(P\ S\ \phi(N))$ (here we do not request that $P \equiv \phi(N)$).

Then (B,S) is called an *m-extension* of (A,R) (m like monomorphism!).

(B,S) is called a *conservative* m-extension of (A,R) if:

$$\forall a,\, b \in A\,,\,(\phi(a) =_S \phi(b)\ \Leftrightarrow a =_R b)$$

The proof of the following proposition is trivial:

Proposition 1.1 *A confluent m-extension (B,S) of (A,R) is conservative.*

Term Rewriting Systems

Now we suppose that E is a first-order algebra $T_{\mathcal{F}}(V)$ where V is a set of variables and $\mathcal{F}$ is the signature of this algebra.
Let $M \in T_{\mathcal{F}}(V)$. $O(M)$ denotes the set of occurrences of M and $M|_u$ is the sub-term of M at the occurrence u.

Definition 1.3 A relation R on $T_{\mathcal{F}}(V)$ is:

1. *stable* if, for any substitution σ, if $(M\ R\ N)$ then $(\sigma(M)\ R\ \sigma(N))$.
2. *compatible* if, for any term P, for any $u \in O(P)$, if $(M\,R\,N)$ then $(P[u \longleftarrow M]\,R\,P[u \longleftarrow N])$.

Definition 1.4 A *rewriting system* is a finite set C of couples $(g_i\,,\,d_i)$ of terms in $T_{\mathcal{F}}(V)$, such that $V(d_i) \subseteq V(g_i)$. These couples are called rewriting rules.
The *rewriting relation* induced by C is the smaller stable and compatible relation containing C. It is also denoted by C. $(M\ C\ N)$ will sometimes be denoted by $(M \xrightarrow{C} N)$.
A *redex* is an instance of the left member of a rule. Let M a term such that $M|_u = \sigma(g)$. The term $N = M[u \longleftarrow \sigma(d)]$ is a *reduct* of M. We will use the following notation to specify the redex occurrence: $M \xrightarrow{C}_u N$.

The test for weak confluence may be restricted to certain couples of terms: the critical pairs whose definition is recalled in the following:

Definition 1.5 Let two rules (g, d) and (l, r). Let u be an occurrence of g such that $g|_u$ is not a variable and $g|_u$ and l are unifiable. Let $N = g|_u \vee l$. Let σ be the substitution such that $N = \sigma(g|_u)$ and τ the substitution such that $N = \tau(l)$.
The superposition of these two rules determines the *critical pair* (P, Q) defined by:

$$P = \sigma(g\,[\,u \longleftarrow \tau(r)\,]) \quad Q = \sigma(d)$$

Proposition 1.2 *A rewriting relation C is weakly confluent if and only if, for any critical pair (P, Q) between two rules of C, there exists a term S such that:*

$$P\ C^*\ S \qquad \text{and} \qquad Q\ C^*\ S$$

1.2 λ-calculi

The pure λ-calculus on a set V of variables is denoted by $\mathbf{\Lambda(V)}$. We suppose familiarity with this theory and we only recall de Bruijn notation. for further details, see [1,11,12,16].

Definition 1.6 $\mathbf{\Lambda}$, the set of λ-calculus terms in de Bruijn's notation, is defined inductively as follows:

1. If $n \in \mathrm{N}$, then $n \in \mathbf{\Lambda}$
2. If M and $N \in \mathbf{\Lambda}$, then $M\ N \in \mathbf{\Lambda}$
3. If $M \in \mathbf{\Lambda}$, then $\lambda(M) \in \mathbf{\Lambda}$

Substitution, in De Bruijn's formalism, is defined as follows:

Definition 1.7 The substitution of N at the height n in M, denoted by $\sigma_n(M,N)$, and the incrementation with n from i, denoted by $\tau_i^n(M)$, are defined by induction as follows:

$$\begin{array}{rcl}\sigma_n(M\,P,N) & = & \sigma_n(M,N)\,\sigma_n(P,N)\\ \sigma_n(\lambda.M,N) & = & \lambda(\sigma_{n+1}(M,N))\\ \sigma_n(m,N) & = & \left\{\begin{array}{rl} m-1 & \text{if} \quad m>n\\ \tau_0^n(N) & \text{if} \quad m=n\\ m & \text{if} \quad m<n\end{array}\right.\end{array}$$

where:

$$\begin{array}{rcl}\tau_i^n(M) & = & \left\{\begin{array}{rl} m+n & \text{if} \quad m\geq i\\ m & \text{if} \quad m<i\end{array}\right.\\ \tau_i^n(M\,P) & = & \tau_i^n(M)\,\tau_i^n(P)\\ \tau_i^n(\lambda(M)) & = & \lambda(\tau_{i+1}^n(M))\end{array}$$

Definition 1.8 The β-reduction in Λ is the rewriting relation defined by the rule:

$$(\lambda\,.M)N \longrightarrow \sigma_0(M,N)$$

We now turn to η-reduction.

Definition 1.9 $M \in \Lambda$ verifies the *condition* $C(\eta)$ if and only if:

> for any occurrence u of a number p in M, one has $p \neq (|u|,M)$ where $(|u|,M)$ is the number of λ whose occurrences are prefixes of u, so the height in λ of u.

The *decrementation* operation is defined for any term M verifying $C(\eta)$. It is denoted by $M^{\downarrow}$. $M^{\downarrow}$ is obtained from M by replacing any number p with occurrence u in M by the number $(p-1)$ provided that p verifies:

$$p > (|u|,M)$$

Definition 1.10 The η-reduction in Λ is the rewriting relation defined by the rule:

$$\lambda.A\,0 \longrightarrow A^{\downarrow} \text{ if } A \text{ verifies } C(\eta)$$

Now we give the two classical ways to add couples.

Definition 1.11 The applicative λc- calculus, denoted by $\Lambda_{\mathbf{c,a}}$, is the pure λ-calculus extended with constants D, F, S and rules :

$$\begin{array}{lrcl}(\text{Fst}) & F\,x\,y & \longrightarrow & x\\ (\text{Snd}) & S\,x\,y & \longrightarrow & y\\ (\text{SP}) & D(\,Fx)(\,Sx) & \longrightarrow & x\end{array}$$

Definition 1.12 The functional λc- calculus, denoted by $\Lambda_{\mathbf{c,f}}$, is obtained from the pure λ-calculus by adding a binary operator denoted by $<,>$, two unary operators denoted by *fst* and *snd* and the rules:

$$\begin{array}{lrcl}(\text{Fst}) & fst\,(<x,\,y>) & \longrightarrow & x\\ (\text{Snd}) & snd\,(<x,\,y>) & \longrightarrow & y\\ (\text{SP}) & <fst\,(x)\,,\,snd\,(x)> & \longrightarrow & x\end{array}$$

These rules, with β (resp. $\beta\eta$) rule, define the theory βSP (resp. ($\beta\eta SP$)). The theory βP is obtained from βSP by removing the (SP)-rule. $\Lambda_{\mathbf{c,a}}(\text{V})$ and $\Lambda_{\mathbf{c,f}}(\text{V})$ are the corresponding extensions of $\Lambda(\text{V})$.

$$
\begin{array}{lrcl}
\text{(Ass)} & (x \circ y) \circ z & = & x \circ (y \circ z) \\
\text{(IdL)} & Id \circ x & = & x \\
\text{(IdR)} & x \circ Id & = & x \\
\text{(Fst)} & Fst \circ < x, y > & = & x \\
\text{(Snd)} & Snd \circ < x, y > & = & y \\
\text{(Dpair)} & < x, y > \circ z & = & < x \circ z, y \circ z > \\
\text{(FSI)} & < Fst, Snd > & = & Id \\
\text{(SP)} & < Fst \circ x, Snd \circ x > & = & x \\
\text{(D}\Lambda\text{)} & \Lambda(x) \circ y & = & \Lambda(x \circ < y \circ Fst, Snd >) \\
\text{(Beta)} & App \circ < \Lambda(x), y > & = & x \circ < Id, y > \\
\text{(AI)} & \Lambda(App) & = & Id \\
\text{(S}\Lambda\text{)} & \Lambda(App \circ < x \circ Fst, Snd >) & = & x
\end{array}
$$

Figure 1: The system $CCL\beta\eta SP$

Theorem 1.1 *$(\Lambda_{c,a}$, $\beta P)$,$(\Lambda_{c,f}$, $\beta P)$ verify Church-Rosser Property.*
$(\Lambda_{c,a}$ $\beta SP)$,$(\Lambda_{c,f}$, $\beta SP)$ do not verify Church-Rosser Property

The first part of this result presents no difficulties. The conjecture defined by the second part was stated by Mann during 1972. The first counter-example and, as far as we know, the only one until ours, was found by J. W. Klop [12,13]. Recently R. De Vrijer [19] proved that $\Lambda_{c,f}(V)$ is a conservative extension of $\Lambda(V)$ and, jointly with J.W. Klop [13] that $\Lambda_{c,f}(V)$ has the unique normal form property.

1.3 The Strong Categorical Combinatory Logic

Upon the idea that semantics can be made akin to syntax , P.L. Curien introduced the Categorical Combinatory Logic, called C.C.L. Numerous results about this -typed or untyped- theory may be found in its extensive monography [6]. We recall only the results which are needed in the following.

Definition 1.13 C.C.L. is a first-order algebra. Its signature consists of:

1. two binary operators: the composition "$\circ$" and the pairing $<,>$.
2. one unary operator: the curryfication Λ.
3. four constants: the identity *Id*, the projections *Fst* and *Snd* and the applicator *App*.

Definition 1.14 The Strong Categorical Combinatory Logic,$CCL\beta\eta SP$, is defined by the equations of figure 1.

We recall the DB-translation between $\Lambda_{c,f}(V)$ and C.C.L. designed by P.L. Curien.

Definition 1.15 Let $M \in \Lambda_{\mathbf{c,f}}(V)$. Let $(x_0, \ldots, x_n)$ a list of variables such that $FV(M) \in (x_0, \ldots, x_n)$. The term $M_{DB(x_0,\ldots,x_n)}$ is defined by:

1. If $M = x$, then: $M_{DB(x_0,\ldots,x_n)} = Snd \circ Fst^i$ where i is the smaller integer such that $x = x_i$.
2. If $M = N\ P$, then: $M_{DB(x_0,\ldots,x_n)} = App \circ < N_{DB(x_0,\ldots,x_n)}, P_{DB(x_0,\ldots,x_n)} >$
3. If $M = \lambda x.N$, then: $M_{DB(x_0,\ldots,x_n)} = \Lambda(N_{DB(x,x_0,\ldots,x_n)})$
4. If $M = fst(N)$, then: $M_{DB(x_0,\ldots,x_n)} = Fst \circ N_{DB(x_0,\ldots,x_n)}$
5. If $M = snd(N)$, then: $M_{DB(x_0,\ldots,x_n)} = Snd \circ N_{DB(x_0,\ldots,x_n)}$
6. If $M =< N, P >$, then: $M_{DB(x_0,\ldots,x_n)} =< N_{DB(x_0,\ldots,x_n)}, P_{DB(x_0,\ldots,x_n)} >$

Now the following proposition gives a first connection between the two theories:

Proposition 1.3 *Let M and $N \in \Lambda_{\mathbf{c,f}}$ such that $FV(M) \cup FV(N) \subseteq \{x_0, \ldots, x_n\}$. Then:*

$$M =_{\beta\eta SP} N \implies M_{DB(x_0,\ldots,x_n)} =_{CCL\beta\eta SP} N_{DB(x_0,\ldots,x_n)}$$

$$M \xrightarrow{\beta} N \implies M_{DB(x_0,\ldots,x_n)} \xrightarrow{Beta} \xrightarrow{Subst} N_{DB(x_0,\ldots,x_n)}$$

The β-reduction is firstly simulated by one (Beta)-reduction followed by a derivation with the rules of a sub-system called *Subst* defined below. The research of occurrences concerned with the started substitution is broken in several steps, each one being the passage through one node of the term (see example page 12). Note that (FSI) or (SP) steps can be avoided with a good choice of the strategy. These remarks are the starting point of our study of the sub-systems of *CCLβηSP*.

We now recall the fundamental equational result of P.L. Curien. From the translation $M_{DB(x_0,\ldots,x_n)}$, P.L. Curien defined a translation M_{CCL} from $\Lambda_{\mathbf{c,f}}$ into C.C.L. He also designed a translation denoted by $A_{\lambda c}$ from C.C.L. into $\Lambda_{\mathbf{c,f}}$. As we will not use them, we do not recall these translations but we give the following result:

Theorem 1.2 Curien Theorem of Equivalence
Let A and $B \in$ C.C.L, M and $N \in \Lambda_{\mathbf{c,f}}$. Then:

$$\begin{array}{rcl}
M =_{\beta\eta SP} N & \implies & M_{CCL} = N_{CCL} \\
A =_{CCL\beta\eta SP} B & \implies & A_{\lambda c} = B_{\lambda c} \\
M_{CCL,\lambda c} & =_{\beta\eta SP} & M \\
A_{\lambda c,CCL} & =_{CCL\beta\eta SP} & A
\end{array}$$

In the sequel, we will only work with ground terms: the combinators. So the set of combinators will still be called C.C.L.

1.4 Sub-systems of *CCLβηSP* : weak confluence, termination

The equations of *CCLβηSP* will be always oriented from left to right. We define several interesting sub-systems of *CCLβηSP* in the following:

Definition 1.16 SL = *(Ass) + (IdL) + (IdR) + (Fst) + (Snd) +(Dpair) + (DΛ)*

$SL\beta = SL + (Beta)$ $\qquad\qquad$ $SL\beta\eta = SL\beta + (AI) + (S\lambda)$

$Subst = SL + (FSI) + (SP)$ $\qquad\qquad$ $CCL\beta SP = Subst + (Beta)$

With the system KB, developed at I.N.R.I.A., which contains an implementation of the Knuth-Bendix Algorithm [14], we obtain the following results:

Proposition 1.4 *Subst , CCLβSP are weakly confluent.*
SL , SLβ , SLβη , CCLβηSP are not weakly confluent.

In the above presentation, we have noticed that Λ-calculus substitution may be computed with the sub-system *Subst* , so broken in several steps, each of them being the crossing of a node of the term's tree. Therefore, we have to ensure that this travel may be done in a non-deterministic way to obtain the confluence of *Subst*. As it is weakly confluent, it suffices to prove its termination. This work was done jointly with A. Laville and published in [10].

Theorem 1.3 *Subst is terminating.*

We say only a few words about the proof of this result. The presence of the rules (Ass) and (DΛ) strongly pertubated all attempts to use standard techniques: polynomial orderings, Recursive Path Orderings, Recursive Decomposition Orderings ... We analysed the maximal number of applications of the rule (DΛ) in any derivation of a given term M. This analysis requires as subroutines the analysis of the maximal number of Λ and of $<,>$ in any term N derived from M. The analysis of the pairs is the most tricky: we computed this number as the length of a list which may be viewed as the list of the Λ-heights of the leaves of the tree associated to the "worst" N. We refer to [9,10] for further details.

Remark 1.1 In the above presentation, we stated that the sub-system *CCLβSP* can simulate reductions of $\Lambda_{c,f}$ therefore it cannot be terminating.

2 Confluence Properties for subsystems of *CCLβηSP*

2.1 Statement of the problems

In the previous section, we showed that the confluent sub-system *Subst* can manage the substitution operation and that the β-reduction is calculated with the sub-system *CCLβSP*. This sub-system is weakly confluent. ($\Lambda_{c,f}$, βSP) is not confluent. What about *CCLβSP* 's confluence ? We will prove that *CCLβSP* is not confluent: the Surjective Pairing rule (SP) destroys confluence property.

But, by doing it carefully, we remarked that we get substitution's simulation without using this rule (SP) and its degenerated form (FSI). Therefore we remove these two rules from *Subst* , giving rise to the system *SL* (see figure 2).

SL is not weakly confluent: the term $\Lambda(x) \circ Id$ creates a critical pair between (DΛ) and (IdR). Its resolution needs the rule (FSI). Now there is a critical pair between (FSI) and (Dpair) whose resolution needs (SP):

$$\Lambda(x) \circ Id \overset{(IdR)}{\longrightarrow} \Lambda(x)$$

(Ass)	$(x \circ y) \circ z$	$=$	$x \circ (y \circ z)$
(IdL)	$Id \circ x$	$=$	x
(IdR)	$x \circ Id$	$=$	x
(Fst)	$Fst \circ < x, y >$	$=$	x
(Snd)	$Snd \circ < x, y >$	$=$	y
(Dpair)	$< x, y > \circ z$	$=$	$< x \circ z, y \circ z >$
(DΛ)	$\Lambda(x) \circ y$	$=$	$\Lambda(x \circ < y \circ Fst, Snd >)$

Figure 2: The rewriting system SL

$$\Lambda(x) \circ Id \overset{(D\Lambda)}{\longrightarrow} \Lambda(x \circ < Id \circ Fst, Snd >) \overset{(IdL)}{\longrightarrow} \Lambda(x \circ < Fst, Snd >)$$

$$< Fst, Snd > \circ x \overset{(FSI)}{\longrightarrow} Id \circ x$$

$$< Fst, Snd > \circ x \overset{(Dpair)}{\longrightarrow} < Fst \circ x, Snd \circ x >$$

(SL) without (IdR) is weakly confluent but cannot manage the substitution operation. Furthermore we want to perform the β-reduction. $SL\beta$ (SL + (Beta)) is not weakly confluent. There is one critical pair between (Beta) and (Ass):

$$(App \circ < \Lambda(x), y >) \circ z \longrightarrow^{*} (x \circ < z \circ Id, y \circ z >) \; ; \; (x \circ < z, \, y \circ z >)$$

whose resolution needs (IdR). So we cannot escape this rule (IdR). The only way to get confluence results for SL and $SL\beta$ is to restrict the set of terms: we will define a subset $\mathcal{D}$ of C.C.L. upon which $SL\beta$ is confluent. To obtain these results, we cannot use Newman Lemma: $SL\beta$ has infinite derivations. Furthermore the rule (Ass) is essential to manage substitution. It pertubates all attempts to use a standard technique for confluence. For example there is no way to construct a parallelisation relation R_P (s.t. which may reduce several redexes already present in a term) which verifies:

$$SL\beta \subseteq R_P \subseteq SL\beta^{*}$$

and which is strongly confluent. The proof of this fact is given in the following remark:

Remark 2.1 Let M, N, P, Q be four constants.
Let X be the term $((M \circ N) \circ P) \circ Q$. We have:

$$X \overset{(Ass)}{\longrightarrow} Y \equiv (M \circ (N \circ P)) \circ Q \text{ and } X \overset{(Ass)}{\longrightarrow} Z \equiv (M \circ N) \circ (P \circ Q)$$

therefore:

$$X \quad R_P \quad Y \qquad \text{and} \qquad X \quad R_P \quad Z$$

But Y and Z have only one redex. So the only possibility is:

$$Y \quad R_P \quad M \circ ((N \circ P) \circ Q) \text{ and } Z \quad R_P \quad M \circ (N \circ (P \circ Q))$$

A created redex has to be reduced: R_P cannot be strongly confluent.

So we will build a new method: the Interpretation Method described in the section 2.2 which will be used to obtain all our confluence results.

2.2 The Interpretation Method

First we recall the definition of an interpretation:

Definition 2.1 Let E and F be two sets. Let $\mathcal{E}$ an application from E into F. Let R be an internal relation of E. A $\mathcal{E}$-interpretation of R, $\mathcal{E}(R)$, is an internal relation of F such that the following diagram holds:

$$\begin{tikzcd}
M \arrow[r, "R"] \arrow[d] & N \arrow[d] \\
\mathcal{E}(M) \arrow[r, "\mathcal{E}(R)"] & \mathcal{E}(N)
\end{tikzcd}$$

Our method is based on the following general lemma.

Proposition 2.1 *Let R a reduction relation defined on a set E and $\mathcal{E}$ an internal relation of E such that:*

1. $\mathcal{E} \subset R^\star$
2. *$\mathcal{E}$ is confluent and terminating.*
3. *There exists an interpretation of R, $\mathcal{E}(R)$, such that:*

$$\mathcal{E}(R) \subseteq R^\star$$

Let $X \subseteq E$. Then:
$\mathcal{E}(R)$ is confluent on $\mathcal{E}(X)$ iff R is confluent on X.

Proof
We have only to draw the diagrams of the figures 3 and 4.

How to make use of such a method ?

1) Let R be a rewriting system defined by the rules $r_1, \ldots, r_n$. The relation $\mathcal{E}$ may be defined by a sub-system of R or only by certain instances of certain rules ... For example if R contained a rule

$$
\begin{tikzcd}
M \arrow[rr, "R^\star"] \arrow[dr] \arrow[dd, "R^\star"'] & & N \arrow[d, "R^\star"] \\
& \mathcal{E}(M) \arrow[r, "\mathcal{E}(R)^\star"] \arrow[d, "\mathcal{E}(R)^\star"] & \mathcal{E}(N) \arrow[d, "\mathcal{E}(R)^\star"] \\
P \arrow[r] & \mathcal{E}(P) \arrow[r, "\mathcal{E}(R)^\star"] & \mathcal{E}(Q)
\end{tikzcd}
$$

Figure 3: Confluence of R

$$
\begin{tikzcd}
\mathcal{E}(M) \arrow[r, "\mathcal{E}(R)^\star"] \arrow[d, "\mathcal{E}(R)^\star"'] & \mathcal{E}(N) \arrow[r, phantom, "\equiv"] \arrow[d, "(R)^\star"] & \mathcal{E}(N) \arrow[dd, "\mathcal{E}(R)^\star"] \\
\mathcal{E}(P) \arrow[r, "R^\star"] \arrow[d, phantom, "|||"] & Q \arrow[dr] & \\
\mathcal{E}(P) \arrow[rr, "\mathcal{E}(R)^\star"] & & \mathcal{E}(Q)
\end{tikzcd}
$$

Figure 4: Confluence de $\mathcal{E}(R)$

such (Ass), then this rule would be included in $\mathcal{E}$ in order to handle redex creations. Now let X_i be the sub-set of $\mathcal{E}(E)$ where $\mathcal{E}(r_i)$ is defined. Then the $\mathcal{E}$-interpretation of R is defined on $\cap_i X_i$ and is the union of the $\mathcal{E}$-interpretations of the (r_i). Furthermore if $r_i \subseteq \mathcal{E}$ then $\mathcal{E}(r_i)$ is the identity.

2) Now the construction of $\mathcal{E}(R)$ is done in three stages as follows (see figure 5):

Let M be a term having a r-redex A at the occurrence u and let B be the reduct of A:

$$M[u \longleftarrow A] \quad \xrightarrow{r} \quad M[u \longleftarrow B]$$

On one hand interpret the context: let Ω be an inert constant (a hole), let C be the context $M[u \longleftarrow \Omega]$. The interpretation of C is $C' = \mathcal{E}(C)$. Ω may appear at several occurrences of C'. On the other hand interpret the redex and its reduct. $\mathcal{E}(A)$ and $\mathcal{E}(B)$ are called the "fragments". Now, stick up $\mathcal{E}(A)$ at any occurrence of Ω in C'. We do not get $\mathcal{E}(M)$: this sticking may create new $\mathcal{E}$-redexes. We have to reduce them in order to get $\mathcal{E}(M)$. $\mathcal{E}(N)$ is obtained in the same way. This interpretation is well-chosen if, despite the redex creations, the interpretation of r is well-defined.

If R is not weakly confluent and if however the rules verifying $r_i \not\subseteq \mathcal{E}$ have only critical pairs with the rules verifying $r_j \subseteq \mathcal{E}$ then, by taking their $\mathcal{E}$-normal form, some instances of such critical pairs may disappear. So the interpretation is letting out the "essential" instances of these critical pairs. Now either we add the relation obtained by superposition of these "essential" instances or we restrict the set of terms: we only accept the terms which cannot create such "essential" instances. The construction of the subset $\mathcal{D}$ in the section 2.3 gives an example of this last choice. Furthermore, all the confluence results of this paper are obtained with this interpretation method. In [9], another application of this method may be found. It proves that the system obtained by removing from $SL\beta$ the rule (DΛ) and adding a rule called (Beta') in [6] is confluent upon the whole set of terms of C.C.L. This sub-system can simulate the so-called "weak" β-reduction.

2.3 The sub-system SL is confluent on the subset $\mathcal{D}$

As we said in the section 2, SL contains a critical pair between (IdR) and (DΛ) and SL without (IdR) cannot manage the substitution operation (see the following examples 1 and 2). Instead of simply deleting (IdR) from (SL), we will replace this rule by its two following instances:

$$Fst \circ Id \overset{(FiD)}{\longrightarrow} Fst \qquad Snd \circ Id \overset{(SiD)}{\longrightarrow} Snd$$

It is easy to see that these instances are sufficient to simulate the λ-calculus substitution (see again Examples 1 and 2). The so-obtained system is called $\mathcal{E}$ (see figure 6).

Remark 2.2 The following rule can remove the critical pair between (IdR) and (DΛ) too:

$$\lambda(x) \longrightarrow \lambda(x \circ < Fst, Snd >)$$

but only with this orientation. So it has no operational sense !

Example 1

Let M be the term: $M \equiv App \circ < \Lambda(< \Lambda(< Snd \circ Fst, Snd >), Snd \circ Fst >), Snd >$

Remark that: $M \equiv (\lambda y. < \lambda x. < y, x >, z > z)_{DB\{z\}}$

The substitution is launched by applying the rule (Beta) at the occurrence ϵ:

$$M \longrightarrow^{(Beta)} < \Lambda(< Snd \circ Fst, Snd >), Snd \circ Fst > \circ < Id, Snd > \equiv M'$$

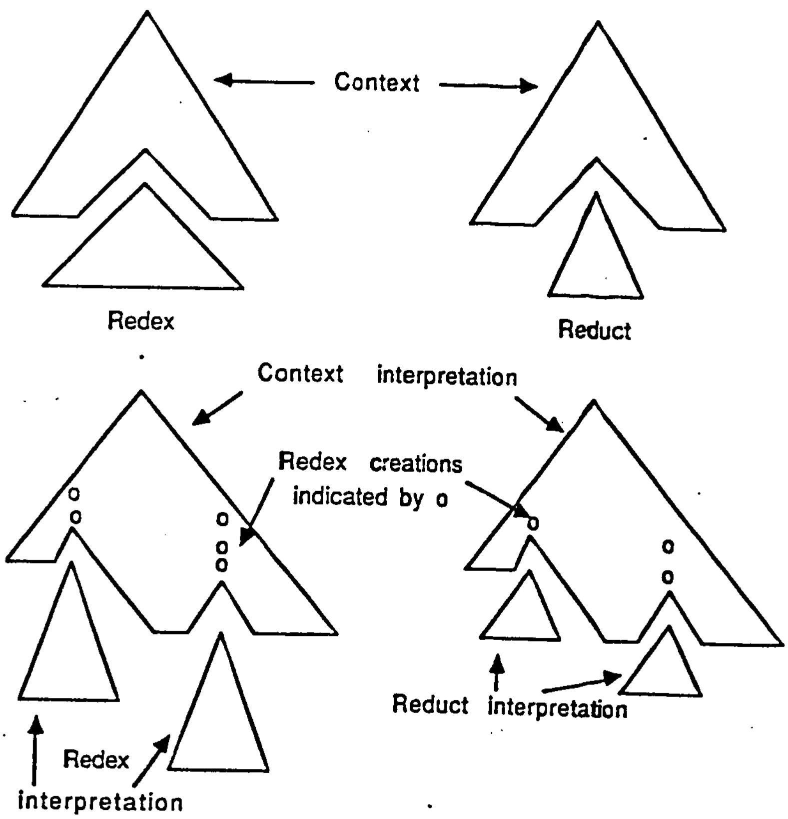


Figure 5: Sticking fragments in a context

The following term is a reduct of M':

$$< \Lambda(< Snd \circ Fst, Snd > \circ << Id \circ Fst, Snd \circ Fst >, Snd >), Snd \circ Id >$$

(use rules (Dpair), (DΛ), (Ass), (Fst) and (Dpair)). We only need the rule (SiD) to obtain the normal form of M'. Now look at the following term:

$$(< \Lambda(< Snd \circ Fst, Snd >), Snd \circ Fst >) \circ Id$$

Here is one of its reducts:

$$< \Lambda(< Snd \circ Fst, Snd > \circ < Id \circ Fst, Snd >), Snd \circ (Fst \circ Id) >$$

Now with (Dpair), (IdL), (Fst) and (Snd), the sub-terms "under" Λ are reduced in normal form: the (IdR)-redex is changed in a (FSI)-redex by an application of the (DΛ)-rule and then may disappear by applications of (Fst) or (Snd) rules. We need the rule (FiD) in order to reduce the term $Snd \circ (Fst \circ Id)$.

Example 2
Let $N \equiv \Lambda(App)$. Note that N is not a DB-translation of a λ-term. $N \circ Id$ has two reducts, itself and $\Lambda(App \circ < Id \circ Fst, Snd >)$ whose (SL)-nomal form is: $\Lambda(App \circ < Fst, Snd >)$. Here we have an "essential" critical pair: to solve it, we have to add either the (FSI)-rule or the following rule:

$$App \circ < Fst, Snd > \longrightarrow App$$

$$
\begin{array}{llcl}
\text{(Ass)} & (x \circ y) \circ z & \longrightarrow & x \circ (y \circ z) \\
\text{(IdL)} & Id \circ x & \longrightarrow & x \\
\text{(Fst)} & Fst \circ < x \, , y > & \longrightarrow & x \\
\text{(Snd)} & Snd \circ < x \, , y > & \longrightarrow & y \\
\text{(Dpair)} & < x \, , y > \circ z & \longrightarrow & < x \circ z \, , y \circ z > \\
\text{(D}\Lambda\text{)} & \Lambda(x) \circ y & \longrightarrow & \Lambda(x \circ < y \circ Fst \, , Snd >) \\
\text{(FiD)} & Fst \circ Id & \longrightarrow & Fst \\
\text{(SiD)} & Snd \circ Id & \longrightarrow & Snd
\end{array}
$$

Figure 6: The system $\mathcal{E}$

which leads to a non-linear rule too.

Now $\mathcal{E}$ is shown weakly confluent with the system KB [14] and as a sub-system of *Subst*, it is terminating. So it can be used to define an interpretation: the interpretation of a term M will be the $\mathcal{E}$-normal form of M denoted by $\mathcal{E}(M)$.

However we want to run β-reduction. So we have to examine the critical pairs between $\mathcal{E}$ and (Beta). There is only one, between (Beta) and (Ass), which is:

$$(\,\mathcal{E}(z \circ Id)\, , \mathcal{E}(z)\,)$$

The rules (FiD) and (SiD) are not sufficient to solve on C.C.L. any instance of this critical pair. By an examination of the $\mathcal{E}$-interpretation of (IdR) upon C.C.L, we will define $\mathcal{D}$, the subset of "well-formed terms": any instance of this critical pair in $\mathcal{D}$ is solved with $\mathcal{E}$.

A–Construction of $\mathcal{D}$

Theorem 2.1 *The system $\mathcal{E}$ (figure 6) is confluent.*

Notations
Let M be a term of C.C.L. $\mathcal{E}(M)$ is the $\mathcal{E}$-normal form of M.
$M(u)$ denotes the symbol of M at the occurrence u. m, n, α, γ often denote sub-words of occurrences. Let u be an occurrence in a term M of C.C.L. The *father* of u is the greater strict prefix of u.

First we describe $\mathcal{E}$-normal forms in order to construct the interpretations of different rewriting relations.

Proposition 2.2 *Let M be a term in $\mathcal{E}$-normal form different from a constant. Then any sub-term of M has one of the following forms where h_i denotes App or Fst or Snd (or Ω the "hole"):*

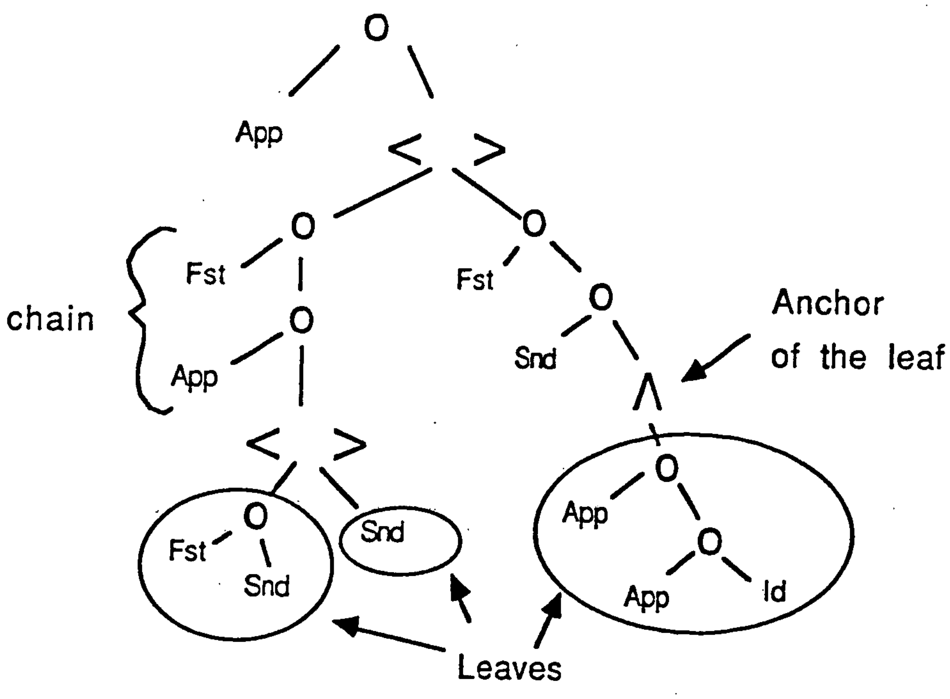


Figure 7: A term in $\mathcal{E}$-normal form

1. $\begin{cases} h_1 \circ (\ldots(h_{n-1} \circ (h_n \circ \Lambda(A))\ldots) & ;\ \Lambda(A) \\ h_1 \circ (\ldots(h_{n-1} \circ (h_n \circ (App \circ < A\,,\,B >)\ldots) & ;\ < A, B > \end{cases}$
where A and B are terms in $\mathcal{E}$-normal form.

2. $k_1 \circ (\ldots(k_{n-1} \circ k_n)\ldots)$ *such that any k_i is a constant. Moreover:* $(\forall i < n)\,,\ k_i \neq Id$ *and* $(k_n = Id) \Longrightarrow k_{n-1} = App$ *(or Ω)*

Proof
Look at the left members of the rules: the top-symbols are only "o". Let u be a given occurrence in a term M in $\mathcal{E}$-normal form. If $M(u)$ is Λ or $<,>$, then $M|_{ui}$ must be in $\mathcal{E}$-normal form, where $i = 0, 1, 2$. If $M(u) =$ "o", then the left son must be a constant different from *Id*. Moreover, if the right son is a pair or the constant *Id*, then the left son cannot be a projection. ■

Definition 2.2 A sub-term of a term in $\mathcal{E}$-normal form built only with compositions and constants, whose father occurrence is Λ or $<,>$, is called a *leaf.* This Λ or this $<,>$ is said to be the *anchor* of the leaf.

So, if a term is represented by a tree with its root as the upper point, its leaves are the maximal sub-terms under the lowest symbols λ or $<\ ,\ >$. Note that the constants h_i of the proposition 2.2 (part 1) are not leaves: such a sequence of constants h_i is called a chain. The extremity of this chain is the symbol Λ or $<,>$ appearing in part 1 of proposition 2.2. All these notations are presented in figure 7.

The leaves of a term will play an essential role: they are the subterms to be examined in order to know if the given term may lead to essential critical pairs, when composed with *Id* on the right.

Definition 2.3 The *height* of a leaf at occurrence u in a term M (in $\mathcal{E}$-normal form) is the number of Λ whose occurrences are prefixes of u. It is denoted by $(|u|, M)$ or simply $|u|$ if no ambiguity.

Notations
We recall Curien's notation [6]:
$\mathcal{P}^0(M) = M$. $\mathcal{P}(M)$ denotes the term $< M \circ Fst\,,\, Snd >$.
$\mathcal{P}^m(M)$ denotes the term $\mathcal{P}(\mathcal{P}^{m-1}(M))$ for $m > 0$.

The following proposition explains how to compute $\mathcal{E}(A \circ B)$ out of $\mathcal{E}(\ A)$ and $\mathcal{E}(\ B)$.

Proposition 2.3 *Let A and B be two terms in C.C.L.. Then:*

$$\mathcal{E}(A \circ B) = \mathcal{E}(A)\,[u_i \longleftarrow \mathcal{E}(F_i \circ \mathcal{P}^{|u_i|}(B))\,]$$

where the u_i are the leaves'occurrences of $\mathcal{E}(A)$. F_i denotes the leaf of $\mathcal{E}(A)$ at the occurrence u_i. The heights $|u_i|$ are measured in $\mathcal{E}(A)$.

Proof
We first prove the following:

$$(A \circ B) \quad \longrightarrow^* \; \mathcal{E}(A)\,[u_i \longleftarrow \mathcal{E}(F_i \circ \mathcal{P}^{|u_i|}(B))\,] \equiv Y$$

by an easy calculation, which can be simulated by waving hands noticing that crossing pairings is mere distribution while crossing Λ increases the $\mathcal{P}$ counter. Now we prove that Y is indeed a $\mathcal{E}$-normal form: any leaf's anchor is a symbol Λ or a pairing so it prevents any redex creation above it. ■

So we obtain the following result:

Corollary 2.1 *Let $M = A \circ Id$. Then:*

$$\mathcal{E}(M) = \mathcal{E}(A)\,[u_i \longleftarrow \mathcal{E}(F_i \circ \mathcal{P}^{|u_i|}(Id))\,]$$

where $\{u_i\} = U$ denotes the S.D.O. of the leaves of $\mathcal{E}(A)$.

The possible critical pair $(A \circ Id, A)$ gives rise to a family of possible essential critical pairs:

$$(\mathcal{E}(\ F_i \circ \mathcal{P}^{|u_i|}(Id))\ ,\ F_i\,)$$

> **A leaf creates an essential critical pair if there exists $m \geq 0$ such that $\mathcal{E}(\ F \circ \mathcal{P}^m(Id))$ is different from F.**

To get rid of these essential critical pairs by adding rules, we should include rules of the form:

$$App \circ \mathcal{P}^m(< Fst\,,\, Snd >) \longrightarrow App$$

in the $\mathcal{E}$-interpretation of (IdR). Remark that $\mathcal{P}(Id) \equiv < Fst\,,\, Snd >$! This interpretation must be included in $(SL)^\star$. So we should add these rules to (SL) and then, because of (Ass) and (Dpair), either the infinite family of rules:

$$App \circ (\mathcal{P}^m(< Fst\,,\, Snd >) \circ x) \longrightarrow App \circ x$$

or the (SP)-rule would be needed. We want to avoid this. Therefore the only solution is to forbid creations of such essential critical pairs by restricting the set of terms. In Examples 1 and 2 page 12, we have noticed that leaves such that $Snd \circ Fst^n$ should not lead to essential critical pairs unlike the leaves Id or App. We define in the following the notion of well-formed leaf: such a leaf should not create essential critical pairs.

Definition 2.4 A leaf F is said *well-formed* if:

$$F \equiv k_1 \circ (\ldots (k_p \circ (Snd \circ Fst^n) \ldots)$$

where $p \geq 0$, $n \geq 0$ and the k_i may be Fst, Snd or App. The "extremity" $Snd \circ Fst^n$ is denoted by $n!$.

Remark 2.3 An ill-formed leaf may have only the following forms:
Id; App; $k_1 \circ (\ldots (k_n \circ App) \ldots)$; Fst^n ; $k_1 \circ (\ldots k_p \circ (App \circ X)$ where $X = Id$ or Fst^n
where the constants k_i may be Fst , App or Snd.

Remark 2.4 If $F \equiv k_1 \circ (\ldots (k_{n-1} \circ k_n) \ldots)$ is a leaf, if M is a term of C.C.L., then, by repetitive use of (Ass), we get:

$$\mathcal{E}(F \circ M) = \mathcal{E}(k_1 \circ (\ldots (k_n \circ M) \ldots))$$

The following proposition asserts that a well-formed leaf does not create an essential critical pair.

Proposition 2.4
1- If F is a well-formed leaf, then:

$$\forall m \geq 0 \,, \; F = \mathcal{E}(F \circ \mathcal{P}^m(Id))$$

2- Conversely if there exists $m_0 \in \mathrm{N}$ such that for all $m \geq m_0$, the leaf F verifies:

$$\mathcal{E}(F \circ \mathcal{P}^m(Id)) = F$$

Then, F is well-formed.
3- Let F be a leaf. If exists $m \in \mathrm{N}$ such that $\mathcal{E}(F \circ \mathcal{P}^m(Id)) \equiv X$ has only well-formed leaves then F is also a well-formed leaf and $X \equiv F$.

Proof
1- Easy calculation noticing that: if A is $\mathcal{E}(Fst^n \circ \mathcal{P}^m(< Fst \,, Snd >))$, then:

$$n \leq m \; : \; A = < \cdots < Fst^{m+1} \,, Snd \circ Fst^m > \cdots > , Snd \circ Fst^n >$$

$$n > m \; : \; A = \mathcal{E}(Fst^{n-m} \circ < Fst^{m+1} \,, Snd \circ Fst^m >) = Fst^n$$

Now we have only to reduce the possible (Snd)-redex.

2,3- Suppose F be ill-formed and compute the left members of the equations. They cannot verify the hypothesis. ■

Now we can describe $\mathcal{D}$ this subset of C.C.L. is such that no term of $\mathcal{D}$ can create essential critical pairs.

Definition 2.5 A term M of C.C.L. belongs to $\mathcal{D}$ iff any leaf of $\mathcal{E}(M)$ is well-formed.

Remark 2.5 Let M a term in $\mathcal{D}$ in $\mathcal{E}$-normal form such that: $M = C[u_i \longleftarrow F_i]$ where the F_i are some leaves of M. Let M_i be some terms of $\mathcal{D}$ in $\mathcal{E}$-normal form. Then:
$C[u_i \longleftarrow M_i] \in \mathcal{D}$ and this term is in $\mathcal{E}$-normal form.

Remark 2.6 Let M be a term of $\mathcal{D}$. In general, subterms of M are not in $\mathcal{D}$.

$\mathcal{D}$ is not stable by the subterm operation.

This lack represents the most important difficulty of our study.

Now we extend to a given term of $\mathcal{D}$ the precedent results for leaves.

Proposition 2.5 *Any term M of $\mathcal{D}$ verifies:*

$$\forall m \geq 0\,,\ \mathcal{E}(M) = \mathcal{E}(M \circ \mathcal{P}^m(Id\ ))$$

Conversely, if there exists $m_0 \geq 0$ such that, for any $m \geq m_0$, M verifies:

$$\mathcal{E}(M) = \mathcal{E}(M \circ \mathcal{P}^m(Id\ )) \text{ then } M \in \mathcal{D}$$

Proof
With the notations and the conclusion of proposition 2.3, we have:

$$\mathcal{E}(M \circ \mathcal{P}^m(Id)\,) = \mathcal{E}(M)[u_i \longleftarrow \mathcal{E}(F_i \circ \mathcal{P}^{|u_i|+m}(Id\ )]$$

where F_i is a leaf of A thus well-formed. With lemma 2.4, we have:

$$\mathcal{E}(F_i \circ \mathcal{P}^{|u_i|+m}(Id\ )) = F_i$$

Conversely, with proposition 2.3, we have:

$$\mathcal{E}(M \circ \mathcal{P}^m(Id\ )) = \mathcal{E}(M)[u_i \longleftarrow \mathcal{E}(F_i \circ \mathcal{P}^{m+|u_i|}(Id\ ))]$$

So any leaf F of $\mathcal{E}(M)$ verifies:

$$F = \mathcal{E}(F \circ \mathcal{P}^{m+|u_i|}(Id\ ))$$

By lemma 2.4, F is well-formed. ∎

Proposition 2.6 *Let M be a term of C.C.L. If exists $m \geq 0$ such that:*

$$\mathcal{E}(M \circ \mathcal{P}^m(Id\ )) \in \mathcal{D} \text{ then } M \in \mathcal{D}$$

Thus, if an (IdR)-redex belongs to $\mathcal{D}$, its reduction does not create essential critical pairs and its reduct also belongs to $\mathcal{D}$. Now we have to extend this result to a term M of $\mathcal{D}$ being a context of an (IdR)-redex $(A \circ Id)$: $M = C[u \longleftarrow A \circ Id]$ where $u \neq \epsilon$. As $\mathcal{D}$ is not stable by the subterm operation, there is no reason for $(A \circ Id)$ to belong to $\mathcal{D}$: look at the following examples

$$Snd \circ < \Lambda(App) \circ Id\,,\ Fst > \in \mathcal{D} \quad (\Lambda(App) \circ Id) \circ Snd \in \mathcal{D}$$

We have to study the $\mathcal{E}$-normal forms of M and of its reduct. To prove our result, we shall use the Interpretation Method. When M is derived to $\mathcal{E}(M)$, this occurrence u may be duplicated, erased ... These transformations of u will become precise with proposition 2.7 on derivations of contexts. Moreover, as explained in the section 2.1, sticking the fragments $\mathcal{E}(A \circ Id)$ and $\mathcal{E}(A)$ in the $\mathcal{E}$-normal forms of the contexts may create new redexes. Proposition 2.8 will give an analysis of these creations.

Interpretation of the contexts

We have to explain how a context is modified when it is derived to its $\mathcal{E}$-normal form and to analyse the splitting of the occurrences of the hole Ω during this derivation. This is done by an examination of the residuals of certain sub-sets of occurrences, called S.D.O., by iterated applications of the (SL)-rules.

Definition 2.6 U is a *set of strictly disjoint occurrences* (in brief a S.D.O.) in a term M if:

$$\forall u, v \in U\ ,(u \neq v \ \Rightarrow\ \exists m\ ,(\ u = m1n \ and\ \ v = m2p\, and\, M(m) = <, >))$$

the pairing at the occurrence m is said to be the *separating pairing* of u and v.
Let $U = \{u_i\}_{i\in[1,n]}$ be a S.D.O. For all i and $j \in [1,n]$, p_{ij} is the occurrence of the separating pairing of u_i and u_j. For all $i \in [1,n]$, $\{p_{ij}\}_{j\in[1,n]}$ is a set of occurrences completely ordered by prefix ordering. Let $p_i = sup_j(p_{ij})$. The pairing at the occurrence p_i is said *the buffer pairing* of u_i. p_i1 or p_i2 is a prefix of u_i. This occurrence is said the *buffer occurrence* of u_i (or occasionally the buffer of u_i).

If the term is represented by a tree, the buffer pairing of u_i is the lowest pairing among the separating pairings of u_i and other occurrences u_j, the buffer occurrence of u_i is the occurrence of the buffer pairing' s son which contains u_i (see figure 8). Note that the set of the buffer occurrences of a S.D.O. is still a S.D.O.

Example: Let M be a term in $\mathcal{E}$-normal form. Then the set of the occurrences of its leaves is a S.D.O. Furthermore, if its anchor is a pairing, the occurrence of a given leaf is its own buffer occurrence.

Remark 2.7 Let U be a S.D.O. in a term M. Suppose that the sub-term of M at the occurrence α is $A \circ B$. Suppose that there exists one occurrence u of U which is $\alpha 1\beta$, then no occurrence of U can be $\alpha 2\gamma$.

Proposition 2.7 *Let C be a context of a S.D.O. $U = \{u_i\}_{i\in[1,p]}$ Suppose that the u_i are marked with " inert constants" Ω_i. Let:*

$$P_0 = C[\,u_i\ \leftarrow\ \Omega_i\,]\,, \forall\, i \in [1,p]$$

For all $n \in$ N, let P_n be such that:

$$P_0 \qquad (SL)^n \quad P_n$$

If Ω_i is a sub-term of P_n, let $\{u_{ij}\}_{j\in[1,k_{i,n}]}$ be the set of the occurrences of this constant in P_n:

$$P_n = C_n\,[u_{ij}\ \leftarrow\ \Omega_i]$$

Let:

$$U_n = \bigcup_{i\in[1,p]} \{u_{ij}\}$$

Then, U_n is a S.D.O.

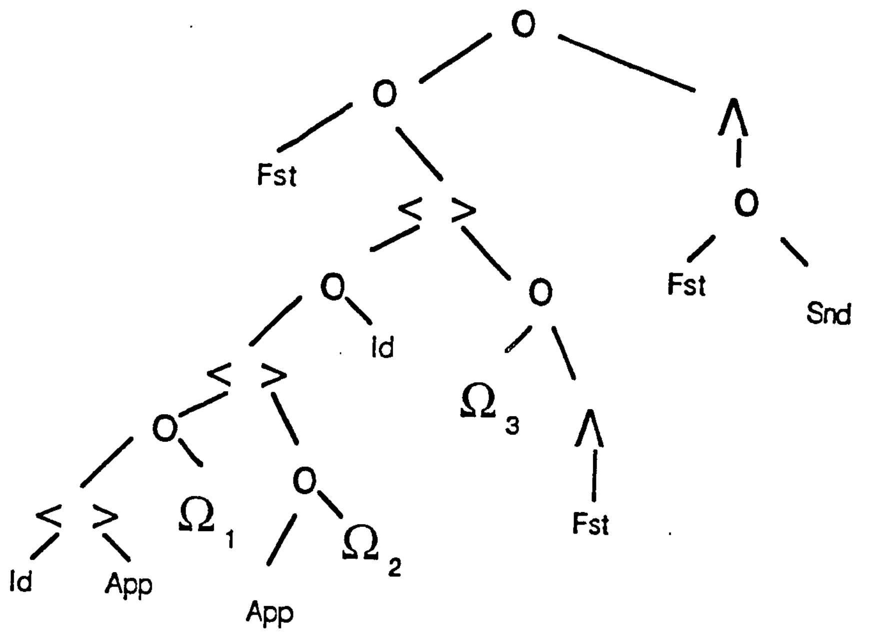


Figure 8: A S.D.O.

Proof
By induction on n. We examine all the possible positions of the occurrence of a given redex in P_n. We observe that the only duplicating rule is (Dpair) and that after one application of this rule the common ancestor of the duplicated residuals is a pairing node. For more details, see [9]. ■

Remark 2.8 *Let $V = \{v_i\}$ be a S.D.O. in a term M. Let p_i be the father of v_i. Let:*

$$u_i = \text{If } (M(p_i) = \circ \text{ and } u_i = p_i 1) \text{ then } p_i \text{ else } u_i$$

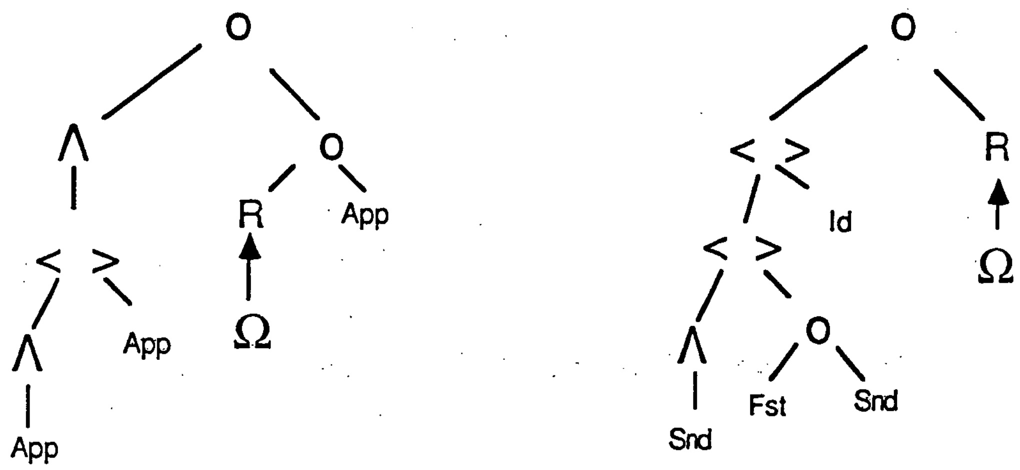
Then, $U = \{u_i\}$ is still a S.D.O.

Sticking the fragments

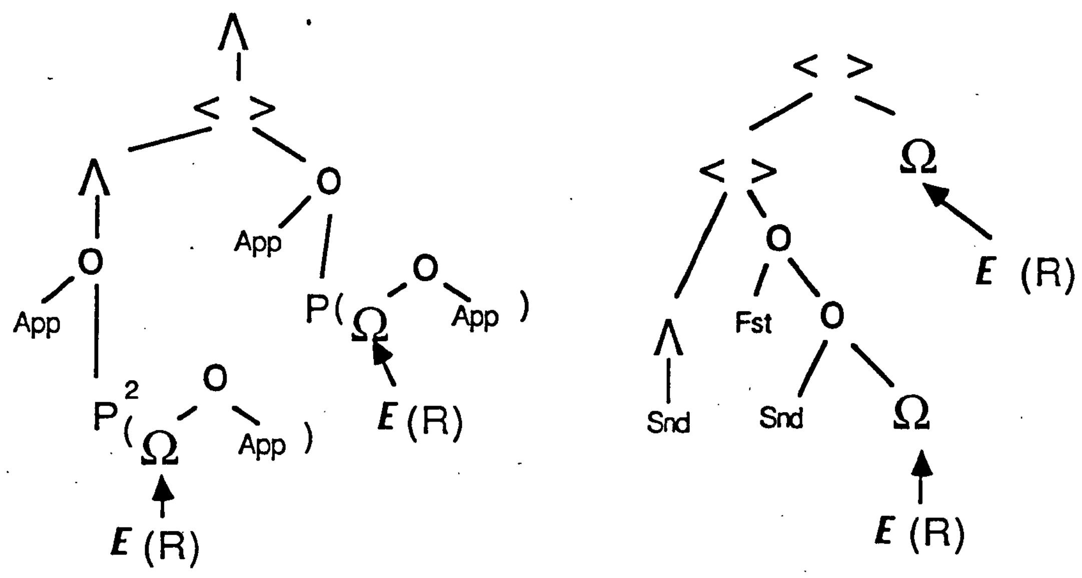
Now we have a first piece of information: an occurrence of a constant Ω in a context in $\mathcal{E}$-normal form may only be (see figure 9):

1. a *son of a* Λ *or a pairing* $<,>$. Sticking a fragment in such an occurrence cannot create a redex.

2. the *right son of a composition.* Then, the context has a leaf such that $k_n = \Omega$ and the other constants in this leaf are different from Ω. Sticking the fragment may create some redexes but these creations cannot get further the anchor of this leaf. Moreover all the created redexes have a constant as left son.



The redex R is replaced by Ω



E normal forms of contexts : Ω will be replaced by E(R)

Figure 9: Interpretation of contexts

3. the *left son of a composition.* Ω is a part of a chain or a part of a leaf: in the context, with the notations of lemma 2.2, there is one constant h_i or one constant k_i (and only one) such that $i \neq n$ which is equal to Ω. A redex, defined by the top-symbol of the fragment, will be created when the fragment is sticking. Its reduction can create other redexes on one hand inside this fragment on the other hand in prefix occurrences as in the precedent case.

The following proposition gives all the informations about the redex creations.

Proposition 2.8 *Let R be a fragment. Let $M = C[u \longleftarrow R]$. Let $P = C[u \longleftarrow \Omega]$. Then:*

1. *There exists a context C' and terms Q_j in $\mathcal{E}$-normal form such that:*

$$\mathcal{E}(P) = C'[u_i \longleftarrow \Omega \ ; \ v_j \longleftarrow \Omega \circ Q_j]$$

such that $\{u_i\} \cup \{v_j\}$ is a S.D.O. Moreover no occurrences u_i are left sons of a composition.

2. *Let M_1 be $C'[u_i \longleftarrow R \ ; \ v_j \longleftarrow R \circ Q_j]$. Redex creations may occur in the sub-terms $R \circ Q_j$. Moreover there may be redex creations in prefix occurrences of u_i and v_j but only if the symbol at their father occurrence is a composition and these last redexes can only be (Fst), (Snd), (FiD) or (SiD) redexes.*
3. *Let p_i be the buffer occurrences of u_i and p_j the ones of v_j. C_p denotes the context of these buffer occurrences in $\mathcal{E}(P)$. Then:*

$$\mathcal{E}(M) = C_p[p_i \longleftarrow \mathcal{E}(P|_{p_i}[\Omega \longleftarrow R]) ; p_j \longleftarrow \mathcal{E}(P|_{p_j}[\Omega \longleftarrow R \circ Q_j])]$$

i.e no creations can appear "above" the buffer occurrences of the S.D.O.

Proof
1– use remark 2.8.
2– Let q_i be the father of u_i. If $C'(q_i)$ is not a composition, then there is no creation of redex in a prefix occurrence of q_i since the context C' is in $\mathcal{E}$-normal form. Else, $u_i = q_i 2$. So the only creations are (Fst), (Snd) -redexes if the top-symbol of the fragment is a pairing and (FiD), (SiD)-redexes if the fragment is *Id*.
Sticking a fragment at occurrences v_j creates a $\mathcal{E}$-redex, defined by the top-symbol of this fragment. The reduction of this redex may create other redexes inside the fragment. There may be also redex creations in a prefix occurrence of v_j but only of (*Fst*), (*Snd*), (FiD) or (SiD)-redexes.
3– By definition, the father of a buffer occurrence is a pair belonging to the context C' in $\mathcal{E}$ -normal form so prevents any redex creation "above" itself. ■

Remark 2.9 Let $M = C[u \longleftarrow R]$. Let $P = C[u \longleftarrow \Omega]$. From any derivation from P to $\mathcal{E}(P)$, one gets a derivation from M to $M_1 = \mathcal{E}(P)[\Omega \longleftarrow R]$.

B– SL is confluent upon $\mathcal{D}$

Now we are able to construct the $\mathcal{E}$-interpretation of the (IdR)-rule on $\mathcal{D}$ expecting that the critical pairs between (DΛ) and (IdR) should disappear.

Proposition 2.9 *Let $M \in \mathcal{D}$ containing a (IdR)-redex at the occurrence u:*
$M = C[u \longleftarrow A \circ Id]$. Let $N = C[u \longleftarrow A]$.
Then, $\mathcal{E}(M) = \mathcal{E}(N)$ and $N \in \mathcal{D}$

Moreover the $\mathcal{E}$-interpretation of (IdR) on $\mathcal{D}$ is the Identity function.

Proof
First we make the fragments appear:
Let $M_1 = C[u \longleftarrow \mathcal{E}(A \circ Id)]$ and $N_1 = C[u \longleftarrow \mathcal{E}(A)]$.
Next we interpret the context. Let P be the context of A in M: $P = C[u \longleftarrow \Omega]$. By proposition 2.8, we know that:

$$\mathcal{E}(P) = C'[u_i \longleftarrow \Omega \circ Q_i \;;\; v_j \longleftarrow \Omega]$$

where $V = \{u_i\} \cup \{v_j\}$ is a S.D.O. Now we stick up the fragments at the occurrences of Ω in $\mathcal{E}(P)$. Using the remark 2.9, we can build, from a derivation from P to $\mathcal{E}(P)$, one derivation from M to M_2 and one another from N to N_2 where:

$$M_2 = C'[u_i \longleftarrow (\mathcal{E}(A \circ Id) \circ Q_i) \;;\; v_j \longleftarrow \mathcal{E}(A \circ Id)]$$

$$N_2 = C'[u_i \longleftarrow \mathcal{E}(A) \circ Q_i \;;\; v_j \longleftarrow \mathcal{E}(A)]$$

We have to reduce all the created $\mathcal{E}$-redexes. We begin by the ones at the occurrences u_i getting the terms M_3 and N_3. Remark that:

$$\mathcal{E}(\mathcal{E}(A \circ Id) \circ Q_i) = \mathcal{E}(A \circ (Id \circ Q_i)) = \mathcal{E}(A \circ Q_i) = \mathcal{E}(\mathcal{E}(A) \circ Q_i)$$

Therefore M_3 and N_3 may only differ by their subterms at their occurrences v_j:

$$M_3 = C''[v_j \longleftarrow \mathcal{E}(A \circ Id)]$$

$$N_3 = C''[v_j \longleftarrow \mathcal{E}(A)]$$

Let q_j be the father of v_j.
If $\mathcal{E}(P)(q_j) = <\,,\,>$ or Λ, then, with proposition 2.8, we can put $\mathcal{E}(A) \circ Id$ in $\mathcal{E}$-normal form without creating redexes in a prefix occurrence of v_j. Therefore $\mathcal{E}(A \circ Id)$ is effectively the sub-term of $\mathcal{E}(M)$ at the occurrence v_j. So its leaves are well-formed. Now:

$$\mathcal{E}(A \circ Id) \equiv \mathcal{E}(\mathcal{E}(A) \circ Id)$$

With proposition 2.6 we conclude that the leaves of $\mathcal{E}(A)$ also are well-formed and that $\mathcal{E}(A \circ Id) = \mathcal{E}(A)$.

If $\mathcal{E}(P)(q_j) = \circ$, then v_j is the right son of a composition. So it is the maximal occurrence of a leaf. Let x_j be the occurrence of this leaf:

$$\mathcal{E}(P)|_{x_j} = k_1 \circ (k_2 \circ (\ldots (k_n \circ \Omega)) \ldots)$$

Sticking $\mathcal{E}(A \circ Id)$ in place of Ω cannot create redexes at a prefix occurrence of x_j because of the leaf's anchor. Therefore, by hypothesis:

$$\mathcal{E}(\; \mathcal{E}(P)|_{x_j}\, [v_j \longleftarrow \mathcal{E}(A \circ Id)\,]\,) \equiv Y \in \mathcal{D}$$

Now since $\mathcal{E}(P)|_{x_j}$ is a leaf, we lift the (IdR)-redex up to x_j by repetitive use of (Ass):

$$Y = \mathcal{E}(\; \mathcal{E}(\; k_1 \circ (k_2 \circ (\ldots (k_n \circ \mathcal{E}(A)) \ldots .)) \circ Id\,)$$

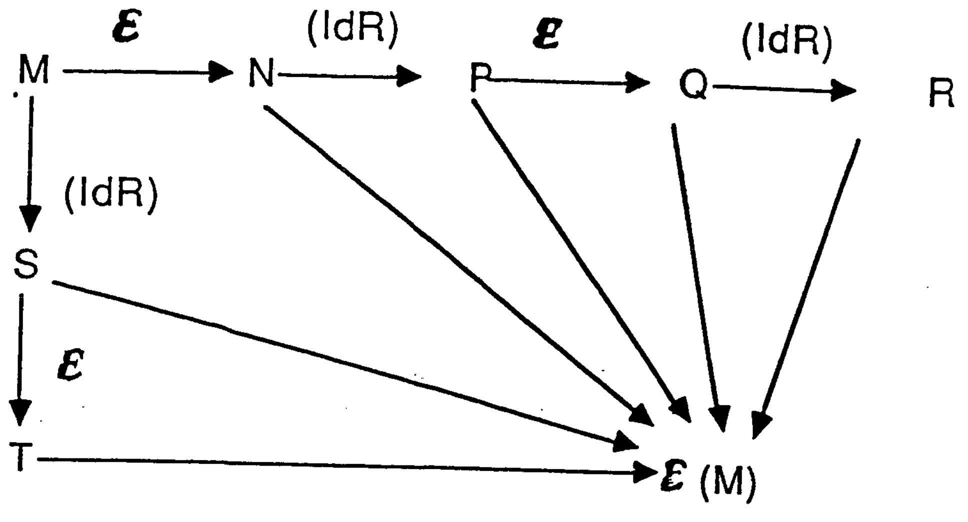


Figure 10: Confluence of (SL) upon $\mathcal{D}$

that is:

$$Y = \mathcal{E}(\ \mathcal{E}(\mathcal{E}(P)|_{x_j}\ [v_j \longleftarrow \mathcal{E}(A)])\ \circ\ Id\,)$$

With the proposition 2.6, we conclude that term $\mathcal{E}(\mathcal{E}(P)|_{x_j}\ [v_j \longleftarrow \mathcal{E}(A)])$ is in $\mathcal{D}$ and is equal to Y. So we have:

$$\mathcal{E}(M) = \mathcal{E}(N)$$

and N belongs to $\mathcal{D}$. ∎

Now we obtain the confluence of (SL) by the aim of the figure 10.

Theorem 2.2 *The rewriting system (SL) is confluent on $\mathcal{D}$.*

$$\forall M \in \mathcal{D}, \qquad SL(M) = \mathcal{E}(M)$$

Notations
If $M \in \mathcal{D}$, then $SL(M)$ denotes the SL-normal form of M.

$$\mathcal{SD} = \{SL(M) \mid M \in \mathcal{D}\}$$

Now there is a simple way to go from C.C.L. into $\mathcal{D}$:

Proposition 2.10 *Let $M \in$ C.C.L. Then, $M \circ Snd \in \mathcal{D}$.*

Proof
Use proposition 2.3 and remark that for any leaf F and for any $m \geq 0$, the term $F \circ \mathcal{P}^m(Snd)$ belongs to $\mathcal{D}$. ∎

2.4 The sub-system $SL\beta$ is confluent on the subset $\mathcal{D}$

As (SL) is confluent upon $\mathcal{D}$ we can manage the substitution in $\mathcal{D}$ but we want to run also the β-reduction in $\mathcal{D}$. We have already made a step by allowing the (IdR)-rule on $\mathcal{D}$: the critical pair between (Ass) and (Beta) is solved on $\mathcal{D}$ as the one between (DΛ) et (IdR). So we only have to prove the confluence of $SL\beta$ on $\mathcal{D}$.

It seems that $SL\beta$ is only a straightforward translation of the relation β of λ-calculus. So we should expect that classical methods of λ-calculus are able to get confluence of $SL\beta$. But as will become precise later the β-reduction corresponds only to the choice of one strategy of $SL\beta$. $SL\beta$ is not terminating. Nevertheless some methods of λ-calculus use strong normalisation of labelled terms. To find such a labelling for C.C.L. terms could be a complicate task: with such a method we should obtain the termination of *Subst* and probably the termination of $SL\beta$ with types. But our examples showing the difficulties for the termination of *Subst* can be typed (see [10]). Another method for λ-calculus is the Axiomatic Method of Tait and Martin-Löf: the relation defined by the reduction of some redexes using an innermost strategy is shown to be strongly confluent. But (Ass) is a rule of $SL\beta$. As we saw in the section 2, there exists no parallelisation of $SL\beta$ which is strongly confluent and the same example shows that a relation based on an innermost strategy cannot be strongly confluent.

Therefore we shall construct the $\mathcal{E}$-interpretation of $SL\beta$. It will be the relation $(Sim\beta)^{\star}$. Since we know that: $SL\beta$ is confluent on $\mathcal{D}$ iff $(Sim\beta)$ is confluent on $\mathcal{SD}$ (proposition 2.1 of the section 2.2), we only have to get the confluence of $(Sim\beta)$ to obtain the result for $SL\beta$. This is an easier problem: terms in $\mathcal{SD}$ are "regularised" terms, whose shapes are very closed to those of $\Lambda_{c,f}$-terms because their leaves are well-formed: proving the confluence of $(Sim\beta)$ can be made with an axiomatic method, inspired by the one of Tait and Martin-Löf.

A – The interpretation of $SL\beta$

First we have to prove that (Beta) is internal to $\mathcal{D}$ and then to build its interpretation.

Definition 2.7 $(Sim\beta)$ is defined on $\mathcal{SD}$ by:

$$M \quad (Sim\beta) \quad N \qquad If \quad M \longrightarrow_{(Beta)} N_1 \quad and \quad N = SL(N_1)$$

Therefore an application of $(Sim\beta)$ consists of: firstly to perform a (Beta)-reduction secondly to carry out the so launched substitution. If the (Beta)-reduction is internal to $\mathcal{D}$, then $(Sim\beta)$ will be well-defined.

Proposition 2.11 *Let $M \in \mathcal{D}$, containing a (Beta)-redex at the occurrence u:*

$$M = C[u \longleftarrow App \circ < \Lambda(A)\,,\, B >]$$

then :

$$N = C[u \longleftarrow A \circ < Id\,,\, B >] \in \mathcal{D}$$

Moreover:

$$SL(M) \quad (Sim\beta)^{*} \quad SL(N)$$

Proof
Let $P = C[u \longleftarrow \Omega]$ be the context of this (Beta)-redex in M. With proposition 2.8, we get:

$$\mathcal{E}(P) = C'[u_i \longleftarrow \Omega \circ Q_i \; ; \; v_j \longleftarrow \Omega]$$

Now, by deriving M and N the fragments appear:

$$M_1 = C[u \longleftarrow \mathcal{E}(App \circ < \Lambda(A) \, , \, B >)]$$

$$N_1 = C[u \longleftarrow \mathcal{E}(A \circ < Id \, , \, B >) \,]$$

From a derivation from P to $\mathcal{E}(P)$, with proposition 2.8, one gets a derivation from M_1 to M_2 and one another from N_1 to N_2:

$$\begin{aligned} M_2 \;\; = C'[u_i \;\; & \longleftarrow \mathcal{E}(App \circ < \Lambda(A), B >) \circ Q_i \\ v_j \;\; & \longleftarrow \mathcal{E}(App \circ < \Lambda(A), B >)] \end{aligned}$$

$$\begin{aligned} N_2 \;\; = C'[u_i \;\; & \longleftarrow \mathcal{E}(A \circ < Id, B >) \circ Q_i \\ v_j \;\; & \longleftarrow \mathcal{E}(A \circ < Id, B >)] \end{aligned}$$

As the patterns $App \circ < \, , \, >$ prevent any redex creation in a prefix occurrence of u_i or v_j, the following term is the SL-normal form of M:

$$\begin{aligned} SL(M) \;\; = C'[u_i \;\; & \longleftarrow App \circ < \Lambda(\mathcal{E}(A \circ \mathcal{P}(Q_i))) \, , \, \mathcal{E}(B \circ Q_i) > \\ v_j \;\; & \longleftarrow App \circ < \Lambda(\mathcal{E}(A)) \, , \, \mathcal{E}(B) >] \end{aligned}$$

Now we construct the $\mathcal{E}$-normal form of N and first, the following reduct of N:

$$\begin{aligned} N_3 \;\; = C'[u_i \;\; & \longleftarrow \mathcal{E}(A \circ < Q_i \, , \, B \circ Q_i >) \\ v_j \;\; & \longleftarrow \mathcal{E}(A \circ < Id \, , \, B >)] \end{aligned}$$

Then by proposition 2.8, if the symbol at the father occurrence of u_i or v_j is a composition, N_3 contains possible (Fst), (Snd), (FiD) or (SiD)-redexes but only "under" the buffer occurrences of u_i and v_j. In order to study these creations, we make these buffer occurrences appear with the following notations:
if $k \in [1, n+p]$ then:

$$\begin{array}{lcl} \alpha_k & = & if \;\; k \leq n \;\; then \quad u_k \;\; else \;\; v_{k-n} \\ w_k & = & the\ buffer\ occurrence\ of\ \alpha_k \end{array}$$

$$\begin{array}{lclr} A_k & = & if \;\; k \leq n \;\; then & SL(A \circ \mathcal{P}(Q_k)) \;\; else \;\; SL(A) \\ B_k & = & if \;\; k \leq n \;\; then & SL(B \circ Q_k) \;\; else \;\; SL(B) \\ Q_k & = & if \;\; k \leq n \;\; then & SL(Q_k) \;\; else \;\; Id \\ T_k & = & SL(P)|_{w_k} & \end{array}$$

With these notations, we have:

$$SL(M) = C'[\alpha_k \longleftarrow App \circ < \Lambda(A_k), B_k >]$$

and:

$$N_3 = C'[\alpha_k \longleftarrow \mathcal{E}(A \circ < Q_k, B_k >) \,]$$

As $M \in \mathcal{D}$, we have:

$$\forall k \in [1, n+p]\,,\ A_k \in \mathcal{SD} \qquad B_k \in \mathcal{SD}$$

With parts 2 and 3 of following lemma 2.2, we get:

$$\forall k \in [1, n+p]\,,\ \mathcal{E}(A \circ < Q_k\,,\, B_k >) \in \mathcal{SD}$$

Now we use proposition 2.8 to study redex creations. Let q_k be the father occurrence of α_k.

1. $\mathcal{E}(P)(q_k)$ is $<\,,\,>$ or Λ: then we replace Ω by and element of $\mathcal{SD}$ therefore:

$$\mathcal{E}(T_k[\Omega \longleftarrow \mathcal{E}(A \circ < Q_k, B_k >)]\,) \in \mathcal{SD}$$

2. $\mathcal{E}(P)(q_k) = \circ$. Then α_k is the maximal occurrence of a leaf F. Let x_k be the occurrence of F. We have: $w_k \leq x_k \leq q_k \leq \alpha_k$. Let F be:

$$F \equiv T_k|_{x_k} \equiv c_1 \circ (c_2 \circ (\ldots (c_n \circ \Omega) \ldots)$$

With lemma 2.1, we obtain that:

$$\mathcal{E}(c_1 \circ (c_2 \circ (\ldots (c_n \circ \mathcal{E}(A \circ < Q_k\,,\, B_k >) \ldots)) \in \mathcal{SD}$$

As the anchor of F prevents any creation of redexes in a prefix occurrence of x_k, we obtain the following:

$$\mathcal{E}(T_k[\Omega \longleftarrow \mathcal{E}(A \circ < Q_k, B_k >)]\,) \in \mathcal{SD}$$

since, by the buffer's definition, there exists one and only one occurrence of Ω in T_k.

Furthermore:

$$\mathcal{E}(N) = C'[w_k \longleftarrow \mathcal{E}(T_k[\Omega \longleftarrow \mathcal{E}(A \circ < Q_k, B_k >)]\,)\,]$$

So we may conclude:

$$\mathcal{E}(N) \in \mathcal{SD}$$

Now we have to build the interpretation of $SL\beta$. In $SL(M)$, the fragment coming from the (Beta)-redex may be modified, duplicated but the occurrences of such modifications are "well separated" since they are strictly disjoint. Furthermore these modifications still are (Beta)-redexes: we have

$$SL(M) = C'[w_k \longleftarrow T_k[\Omega \longleftarrow App \circ < \Lambda(A_k), B_k >]\,]$$

Now, since $N \in \mathcal{D}$, we have:

$$SL(N) = C'[w_k \longleftarrow SL(T_k[\Omega \longleftarrow A \circ < Q_k, B_k >]\,)\,]$$

The interpretation of (Beta) is obtained by successively applying the relation $(Sim\beta)$ at each occurence α_k. Let M_k for $k \in [0, n+p]$ be the term:

$$M_o = SL(M) \qquad M_{k-1} \quad (Sim\beta) \quad M_k$$

the k^{nth} application of $(Sim\beta)$ being intended to reduce the (Beta)-redex at occurrence α_k so that:

$$\begin{aligned} M_k \; = C'[w_s \; & \longleftarrow SL(T_s[\Omega \longleftarrow A_s \circ < Id, B_s >]) \\ w_r \; & \longleftarrow T_r[\Omega \longleftarrow App \circ < \Lambda(A_r), B_r >] \,] \end{aligned}$$

where $s \in [1, k]$. we prove that M_k is in $\mathcal{SD}$ in the same way as we proved that $N \in \mathcal{D}$. Remark that $SL(A_s \circ < Id, B_s >) = SL(A \circ < Q_s, B_s >)$ by the part 3 of the following lemma 2.2. Furthermore:

$$M_{n+p} = C'[w_k \longleftarrow SL(T_k[\Omega \longleftarrow SL(A_k \circ < Id, B_k >)]\,)\,]$$

Therefore:

$$M_{n+p} \equiv SL(N) \text{ and } SL(M) \; (Sim\beta)^{n+p} \; SL(N) \qquad \blacksquare$$

Remark 2.10 This proposition becomes false if one uses the relation *Subst* instead of the relation SL as seen with the following example. Let P be a (Beta)-redex in *Subst*-normal form and Q its reduct. Let $M = < Fst \circ P, Snd \circ P >$.
N is obtained by reducing the (Beta)-redex P in the left son of M.

$$Subst(M) = P \qquad Subst(N) = < Subst(Fst \circ Q), Snd \circ P >$$

We do not have the following: $Subst(M) \; (Sim\beta)^* \; Subst(N)$.

Here are the needed lemmas by the previous theorem:

Lemma 2.1 *If $M \in \mathcal{D}$, then $M \circ \mathcal{P}^m(Fst^n) \in \mathcal{D}$, for all $m, n \geq 0$.*
If F is a leaf, If $M \in \mathcal{D}$, then $F \circ M \in \mathcal{D}$.

Proof
Using the proposition 2.3, we have:

$$\mathcal{E}(M \circ \mathcal{P}^m(Fst^n)) = SL(M)[u_i \longleftarrow F_i \circ \mathcal{P}^{|u_i|+m}(Fst^n)]$$

The result follows, by an easy calculation, noticing that F_i is a well-formed leaf.

Lemma 2.2 1. *If F is a well-formed leaf, if $m \geq 0$, then :*

$$\text{If } B \in \mathcal{D} \;\; , \textit{then} \;\; F \circ \mathcal{P}^m(< Id, B >) \in \mathcal{D}$$

2. *Let A and $B \in \mathcal{D}$. then, for all $m \geq 0$:*

$$X = A \circ \mathcal{P}^m(< Id\, , B >) \in \mathcal{D}$$

3. *let A, B, Q be terms in C.C.L. If:*

$$(A \circ \mathcal{P}(Q)) \in \mathcal{D} \, ; \, (B \circ Q) \in \mathcal{D}$$

then :

$$Y = A \circ < Q\, , B \circ Q > \in \mathcal{D}$$

Furthermore:

$$\mathcal{E}(Y) = \mathcal{E}((A \circ \mathcal{P}(Q)) \circ < Id, B \circ Q >)$$

Proof

1. Easy calculation using the following result. Let $F \equiv K \circ (Snd \circ Fst^n)$. We get :

$$SL(F \circ \mathcal{P}^m(< Id\, , B >)) =$$

   - If $n < m$, then $K \circ (Snd \circ Fst^n)$
   - If $n > m$, then $K \circ (Snd \circ Fst^{n-1})$
   - If $n = m$, then $SL(K \circ (B \circ Fst^m))$

   As $B \circ Fst^m \in \mathcal{D}$ (lemma 2.1) we get $SL(K \circ (B \circ Fst^m)) \in \mathcal{D}$.

2. By lemma 2.5, part 1 of this lemma and the following equality:

$$\mathcal{E}(X) = \mathcal{E}(A)[u_i \longleftarrow \mathcal{E}(F_i \circ \mathcal{P}_A^{|u_i|+m}(< Id\, , B >))]$$

3. With the second part of this lemma, we get:

$$X \equiv (A \circ \mathcal{P}(Q)) \circ < Id, B \circ Q > \in \mathcal{D}$$

   Now the following term Y is a $\mathcal{E}$-reduct of X:

$$Y \equiv A \circ < Q \circ Id, B \circ Q >$$

   Here is the crucial point: Y **contains a (IdR)-redex.** Using proposition 2.9, we get:

$$A \circ < Q, B \circ Q > \in \mathcal{D} \qquad \blacksquare$$

We conclude with the following theorems:

Theorem 2.3 *Let $N \in \mathcal{D}$.*

$$\text{If } N \quad SL\beta^* \quad P \quad \text{, then} \quad SL(N) \quad (Sim\beta)^* \quad SL(P)$$

Proof
By the following diagram:

$$\begin{tikzcd}
M \arrow[r, "(SL)^*"] \arrow[d, "(SL)^*"'] & N \arrow[r, "(Beta)"] \arrow[d] & P \arrow[d, "(SL)^*"] \\
SL(M) \arrow[r, phantom, "\equiv"] & SL(N) \arrow[r, "Sim\beta^*"] & SL(P)
\end{tikzcd}$$

Theorem 2.4 *$SL\beta$ is confluent on $\mathcal{D}$.*

$$
\begin{tikzcd}
M \arrow[rr, "SL\beta^{\star}"] \arrow[dr] \arrow[dd, "SL\beta^{\star}"'] & & N \arrow[d] \\
& SL(M) \arrow[r, "(Sim\beta)^{\star}"] \arrow[d] & SL(N) \arrow[d] \\
P \arrow[r] & SL(P) \arrow[r, "(Sim\beta)^{\star}"] & SL(Q)
\end{tikzcd}
$$

Figure 11: Confluence of $SL\beta$

Proof
The interpretation $(Sim\beta)$ on $\mathcal{SD}$ of $SL\beta$ will be proved confluent below. We can conclude with the diagram 11 ∎

B–Confluence of $(Sim\beta)$

We define the relation $\mathcal{B}$ as an iteration of $(Sim\beta)$ based on an innermost strategy:

Definition 2.8 *$\mathcal{B}$ is defined on $\mathcal{SD}$ by induction as follows:*

$$
\begin{array}{lrcl}
(1) & M & \mathcal{B} & M \\
 & \text{If } M_i \;\; \mathcal{B} \;\; N_i \text{ , then :} & & \\
(21) & < M_1, M_2 > & \mathcal{B} & < N_1, N_2 > \\
(22) & App \circ M & \mathcal{B} & App \circ N \\
(23) & Fst \circ M & \mathcal{B} & SL(Fst \circ N) \\
(24) & Snd \circ M & \mathcal{B} & SL(Snd \circ N) \\
(25) & \Lambda(M) & \mathcal{B} & \Lambda(N) \\
(3) & App \circ < \Lambda(M_1), M_2 > & \mathcal{B} & SL(N_1 \circ < Id, N_2 >)
\end{array}
$$

Proposition 2.12 *$\mathcal{B}$ is internal to $\mathcal{D}$ and verifies:*

$$(Sim\beta) \subseteq \quad \mathcal{B} \quad \subseteq (Sim\beta)^{*}$$

Proof
by first giving an axiomatic version of the definition of $(Sim\beta)$ and then by induction on the length of the proof tree of $(M \;\; \mathcal{B} \;\; N)$. ∎

Theorem 2.5 *$\mathcal{B}$ is strongly confluent.*

Proof
Let $M \; \mathcal{B} \; Q$ and $M \; \mathcal{B} \; P$. We search N such that $Q \; \mathcal{B} \; N$ and $P \; \mathcal{B} \; N$. The proof is made by induction on the length of the proof tree of $M \; \mathcal{B} \; Q$, for all P. This proof needs the two following lemmas.

Lemma 2.3 *If* $M \quad \mathcal{B} \quad N$*, then, for all* $m \geq 0$*, for all* $n \geq 0$*, one has:*

$$SL(M \circ \mathcal{P}^m(Fst^n)) \qquad \mathcal{B} \qquad SL(N \circ \mathcal{P}^m(Fst^n))$$

The second lemma is called Substitution lemma by reference to the substitution lemma of λ-calculus.

Lemma 2.4 *Substitution Lemma of* $SL\beta$
If $M \quad \mathcal{B} \quad N$*, if* $P \quad \mathcal{B} \quad Q$ *then, for any* m*, we have:*

$$SL(M \circ \mathcal{P}^m(< Id, P >)) \qquad \mathcal{B} \qquad SL(N \circ \mathcal{P}^m(< Id, Q >))$$

The proofs of these lemmas are made by induction on the length of the proof tree of ($M \quad \mathcal{B} \quad N$), for all P, Q and any proof tree of ($P \quad \mathcal{B} \quad Q$) , for all m. For more details, see [9]. ■

Using proposition 2.12 and theorem 2.5, we get:

Theorem 2.6 $(Sim\beta)$ *is confluent on* $\mathcal{D}$.

2.5 The sub-system $SL\beta\eta$ is confluent on the subset $\mathcal{D}$

The following rules (AI) and (SΛ) are now added to $SL\beta$:

$$\begin{array}{llcl}
(AI) & \Lambda(App) & \longrightarrow & Id \\
(S\Lambda) & \Lambda(App \circ < x \circ Fst, Snd >) & \longrightarrow & x
\end{array}$$

The system so obtained is called $SL\beta\eta$.

As we recalled in the section 1.3 , P.L. Curien showed that the theories $CCL\beta\eta\mathcal{SP}$ (that is $SL\beta\eta$ $+(SP)$) and $\beta\eta\mathcal{SP}$ are equationally equivalent. This result needs the rules (AI) and (SΛ) which in a certain sense (see below) have to do with the η-rule. Moreover in typed C.C.L., these rules assert the uniqueness of the exponentiation in Cartesian Closed Categories.

The system $CCL\beta\eta\mathcal{SP}$ is not weakly confluent. Therefore we only examine the $\mathcal{E}$-interpretation of the rewriting relation associated with (AI) and (SΛ) on the subset $\mathcal{D}$ of C.C.L. It is called $(Sim\eta)$. The relation $(Sim\eta) \cup (Sim\beta)$ will be proved confluent on $\mathcal{SD}$ and so is the system $SL\beta\eta$ on $\mathcal{D}$.

A–Interpretation of (AI) and (SΛ)

We replace the expression "the leaf F at the occurrence u" by $(F; u)$.

Definition 2.9 A leaf $(F; u)$ in a term M in SL-normal form, is said to be *η-accessible* in M if, $n!$ being its extremity, $n = (|u|, M)$. If $n \geq (|u|, M)$, F is said *free* in M.
The leaf $F^{\downarrow}$ is obtained by replacing F's extremity $n!$ by $(n-1)!$. The leaf F is said to be *decremented.* This operation can be performed only if $n \geq 1$.

Example
In the term $\Lambda(Snd)$, the height of the leaf Snd is 1. It is not accessible. let N be the term:

$$N \equiv \Lambda(\Lambda(< Snd \circ Fst^2, Snd \circ Fst^3 >))$$

The leaf $Snd \circ Fst^2$ is η-accessible ; the leaf $Snd \circ Fst^3$ is free and is not η-accessible.

Definition 2.10 A term M of $\mathcal{SD}$ verifies the *condition* $C(\eta)$ if it has no η-accessible leaves. The term $M^{\downarrow}$ is obtained from M by decrementing all the free leaves of M.

Remark 2.11 Let M be a term of $\mathcal{SD}$ verifying the condition $C(\eta)$. Then the extremities $n!$ of the free leaves of M verify $n \geq 1$ so the term $M^{\downarrow}$ is well-defined as we have:

1. $n \geq (|u|, M)$ since F is free in M.
2. $n \neq (|u|, M)$ since F is not η-accessible.

so $n > (|u|, M) \geq 0$ and then $n \geq 1$.

The following relation called $(S\Lambda_{SD})$ justifies this terminology: the condition $C(\eta)$ is the equivalent of the condition $x \notin FV(M)$ for the η-rule of λ-calculus.

Definition 2.11 The relation $(S\Lambda_{SD})$ is the compatible closure of the relation (still called $(S\Lambda_{SD})$) defined on $\mathcal{SD}$ as follows:

If M verifies the condition $C(\eta)$ then:

$$\Lambda(App \circ < M, Snd >)\ (S\Lambda_{SD})\ M^{\downarrow}$$

If $(M\ \ (S\Lambda_{SD})\ \ N)$, then M is said to *contain a* $S\Lambda_{SD}$*-redex* and N is said to be its *reduct.* In general N is not an element of $\mathcal{SD}$. We will prove that it belongs to $\mathcal{D}$. The relation performing first the reduction of a $S\Lambda_{SD}$-redex then putting the reduct in $\mathcal{E}$- normal form is called $(Sim\eta)$ and is defined as follows:

Definition 2.12 The relation $(Sim\eta)$ is defined on $\mathcal{SD}$ as follows:

$M \overset{(Sim\eta)}{\longrightarrow} N$ if there exists N_1 $\in$C.C.L. such that:

$$M \overset{(S\Lambda_{SD})}{\longrightarrow} N_1 \qquad \text{and} \qquad N = \mathcal{E}(N_1)$$

$(Sim\eta)$ will be shown to be the interpretation of the rewriting relation defined by the rules (AI) and (SΛ) on $\mathcal{D}$. The following lemma gives the key point of the following proofs.

Lemma 2.5 If $M \circ Fst \in \mathcal{D}$, then $M \in \mathcal{D}$. Furthermore, $SL(M \circ Fst)$ verifies the condition $C(\eta)$.

Proof
Suppose M in $\mathcal{E}$-normal form . With the proposition 2.3, we have:

$$SL(M \circ Fst) = M[u_i \longleftarrow SL(F_i \circ \mathcal{P}^{(|u_i|,M)}(Fst))]$$

where the $(F_i; u_i)$ are the leaves of M.
It suffices to prove that, if the leaf F_i is ill-formed, then the leaves of the term $SL(F_i \circ \mathcal{P}^{(|u_i|,M)}(Fst))$ are also ill-formed. This is done by a simple calculation.
Now let $n!$ be the extremity of the leaf $(F; u)$ in M. The extremity of the corresponding leaf in $SL(M \circ Fst)$ is:

1. if $n < (|u|, M)$ then:

$$SL(F \circ \mathcal{P}^{(|u|,M)}(Fst)) = n!$$

2. If $n \geq (|u|, M)$ then:

$$SL(F \circ \mathcal{P}^{(|u|,M)}(Fst)) = n + 1!$$

In this case F is free in M.

As $(|u|, M) = (|u|, SL(M \circ Fst))$, no leaf of $SL(M \circ Fst)$ can be η-accessible. ■

Proposition 2.13 *Let M be a term of $\mathcal{D}$ containing a (AI)-redex. or a (SΛ)-redex. Let N be its reduct:*

$$M = C[u \longleftarrow \Lambda(App)] \quad N = C[u \longleftarrow Id]$$

or

$$M = C[u \longleftarrow \Lambda(App \circ < A \circ Fst, Snd >)] \;\; N = C[u \longleftarrow A]$$

Then N is a term of $\mathcal{D}$. Moreover:

$$SL(M) \stackrel{(Sim\eta)^*}{\longrightarrow} SL(N)$$

Proof

We begin with the (AI)-redex. Let $P = C[u \longleftarrow \Omega]$. Then,

$$Q = SL(P) = C'[u_i \longleftarrow \Omega \circ Q_i \,;\, v_j \longleftarrow \Omega]$$

where v_j is never the left son of a composition.
Let M_1 be the following reduct of M:

$$M_1 = C'[u_i \longleftarrow \Lambda(App) \circ Q_i \,;\, v_j \longleftarrow \Lambda(App)]$$

The top-symbol Λ of the fragment prevents any redex creation in the prefix occurrences of u_i and v_j. Therefore:

$$SL(M) = C'[u_i \longleftarrow \Lambda(App \circ < SL(Q_i \circ Fst), Snd >) \,;\, v_j \longleftarrow \Lambda(App)]$$

From the hypothesis $M \in \mathcal{D}$, one deduces firstly that $\{v_j\} = \emptyset$ and then $SL(Q_i \circ Fst) \in \mathcal{SD}$
By lemma 2.5, none of the leaves of these above fragments are η-accessible and moreover the terms Q_i belong to $\mathcal{D}$.

Let p_i be the buffers of the u_i. With the method already used for proposition 2.9 in the section 2.3, we show:

$$SL(M) \stackrel{(Sim\eta)^*}{\longrightarrow} C'[p_i \longleftarrow SL(Q|_{p_i}[u_i \longleftarrow (SL(Q_i \circ Fst)\,)^{\downarrow}]\,)\,]$$

Let X be the right member of the previous equation. Let N_1 be the following reduct of N:

$$N_1 = C'[u_i \longleftarrow Id \circ Q_i]$$

then:

$$\mathcal{E}(N) = C'[p_i \longleftarrow SL(Q|_{p_i}[u_i \longleftarrow SL(Q_i)]\,)\,]$$

As the terms Q_i belong to $\mathcal{D}$, N itself is in $\mathcal{D}$ (lemma 2.8). Now we have to prove the following identity:

$$X \equiv SL(N)$$

It suffices to obtain, for any i, the following identity:

$$SL(Q|_{p_i}[u_i \longleftarrow (SL(Q_i \circ Fst)^{\downarrow})]) \equiv SL(Q|_{p_i}[u_i \longleftarrow SL(Q_i)])$$

Calculations of lemma 2.5 are used for this last equality.

The second point is proved by examining the sub-terms at the buffer occurrences. With the same notations as above, we get:

$$SL(M)|_{p_i} = C'|_{p_i}[u_i \longleftarrow \Lambda(App \circ < SL(A \circ (Q_i \circ Fst)), Snd >)]$$

because:

$$\begin{aligned} &SL(\Lambda(App \circ < SL(A \circ Fst), Snd >) \circ Q_i) = \\ &\qquad \Lambda(App \circ < SL(A \circ (Q_i \circ Fst)), Snd >) \end{aligned}$$

with the relation $(Sim\eta)$, we get:

$$X|_{p_i} = C'|_{p_i}[u_i \longleftarrow SL(A \circ (Q_i \circ Fst))^{\downarrow}]$$

Now:

$$SL(N)|_{p_i} = C'|_{p_i}[u_i \longleftarrow SL(A \circ Q_i)]$$

We conclude with lemma 2.5.
The proof is the same for the occurrences v_j. ∎

B–Confluence of $SL\beta\eta$ on $\mathcal{D}$

First we prove the confluence of the relation $(Sim\eta)$. The following lemma contains the technical points of this proof.

Lemma 2.6 *let $R = C[v \longleftarrow \Lambda(App \circ < T, Snd >)]$. If R and T verify condition $C(\eta)$, then:*

1. *$R^{\downarrow}$ contains a $(S\lambda_{SD})$-redex at the occurrence v.*
2. *let $R_1 = C[v \longleftarrow T^{\downarrow}]$. R_1 verifies the condition $C(\eta)$.*

Proof

1. By definition we have:

$$R^{\downarrow} = C^{\downarrow}[v \longleftarrow \Lambda(App \circ < T^{\downarrow}, Snd >)]$$

A leaf is free in T iff the leaf at the same occurrence is free in $T^{\downarrow}$: let F be a free leaf in T and α its occurrence. Its extremity is $p!$.

$$p \neq (|v021\alpha|, C) \text{ and } p \neq (|\alpha|, T)$$

(a) Suppose $p < (|v021\alpha|, C)$, then F is not free in R so is not modified in $R^{\downarrow}$ and is still free in $T^{\downarrow}$.

(b) $p > (|v021\alpha|, C)$. Then, the extremity of the leaf at the occurrence α in $T^{\downarrow}$ is $p-1!$. As $(|v021\alpha|, C^{\downarrow}) \geq (|\alpha|, T^{\downarrow}) + 1$, we have: $p - 1 > (|\alpha|, T^{\downarrow})$.

If F is not free in T, it verifies $p < (|\alpha|, T)$ so it is also not free in R.

2. let F a free leaf of R_1 (extremity $p!$, occurrence α in R_1).If v is not a prefix of α, then F, as a leaf of C, verifies $C(\eta)$ by hypothesis. Now a leaf of $T^{\downarrow}$ at the occurrence $v\beta$ in R_1 is issued from a leaf of T with occurrence $v021\beta$ in R, and with extremity $q!$.

   (a) $q < (|\beta|, T)$. Then, $q = p$ and $p < (|v\beta|, R_1)$

   (b) $q > (|\beta|, T)$. Then, $p = q - 1$. As:

$$p \neq (|v021\beta|, R) \text{ and } (|v021\beta|, R) = 1 + (|v\beta|, R_1)$$

   so:

$$p - 1 \neq (|v\beta|, R_1) \quad \blacksquare$$

Proposition 2.14 *The relation $(Sim\eta)$ is strongly confluent.*

Proof

let M be a term of $\mathcal{SD}$ with two $(S\lambda_{SD})$-redexes at the occurrences u and v. The only non trivial case is: u is a prefix of v: $v = u\,w$. Let:

$$M = C[u \longleftarrow \Lambda(App \circ < R, Snd >)]$$

$$R = C_1[w \longleftarrow \Lambda(App \circ < S, Snd >)]$$

$$P = C[u \longleftarrow R^{\downarrow}]$$

$$N = C[u \longleftarrow \Lambda(App \circ < SL(C_1[w \longleftarrow S^{\downarrow}]), Snd >)]$$

By the precedent lemma, N still has a $(S\lambda_{SD})$-redex at the occurrence u. Let:

$$Q = SL(C[u \longleftarrow SL(C_1[w \longleftarrow S^{\downarrow}])^{\downarrow}])$$

so:

$$N \overset{(Sim\eta),u}{\longrightarrow} Q$$

We prove the following point:

$$P \overset{(Sim\eta)^{\epsilon},v}{\longrightarrow} Q$$

we have:

$$R^{\downarrow} = C_1^{\downarrow}[w \longleftarrow \Lambda(App \circ < S^{\downarrow}, Snd >)]$$

let $X = C[u \longleftarrow \Omega]$. let p be the occurrence of the leaf of X which contains Ω. Let:

$$X|_p = k_1 \circ (k_2 \circ (\ldots k_n \circ \Omega)\ldots)$$

We get:
$$P|_p = SL(k_1 \circ (k_2 \circ (\ldots k_n \circ C_1^{\downarrow}[w \longleftarrow \Lambda(App \circ < S^{\downarrow}, Snd >)]) \ldots))$$

$$Q|_p = SL(k_1 \circ (k_2 \circ (\ldots k_n \circ SL(C_1[w \longleftarrow S^{\downarrow}])^{\downarrow}) \ldots))$$

Since $R^{\downarrow}$ is a term of $\mathcal{SD}$, the derivation performed to get P contains only applications of projection rules. Therefore, P has at most one occurrence of $App \circ < S^{\downarrow}, Snd >$. If this derivation erases the occurrence w of $C_1^{\downarrow}$, then it is also erased in Q else we have to prove the following equality:

$$C_1^{\downarrow}[w \longleftarrow (S^{\downarrow})^{\downarrow}] \equiv (C_1[w \longleftarrow S^{\downarrow}])^{\downarrow}$$

It is done by a simple calculation. ■

Now we recall Hindley-Rossen Lemma:

Proposition 2.15 *Let R and S be two relations on a set X. If these two relations commute, if they are confluent, then $R \cup S$ is confluent.*

So we have only to prove the following proposition:

Proposition 2.16 *Relations $(Sim\eta)^\star$ and $(Sim\beta)^\star$ commute.*

Proof
This is a consequence of the following property: (see [1][p. 65]):

$$\begin{tikzcd} M \arrow[r, "Sim\eta"] \arrow[d, "Sim\beta"'] & P \arrow[d, "Sim\beta"] \\ N \arrow[r, "Sim\eta^\star"] & Q \end{tikzcd}$$

First we prove that if M verifies $C(\eta)$, then N itself verifies it. This result is given by lemma 2.7. Then we get the existence of the term Q by examining the following cases:

1. the $S\Lambda_{SD}$-redex contains the (Beta)-redex. See lemma 2.8.
2. the "function part" of the (Beta)-redex. contains the $S\Lambda_{SD}$-redex. See lemma 2.9.
3. the "argument part" of the (Beta)-redex. contains the $S\Lambda_{SD}$-redex. See lemma 2.10. ■

Now we may conclude:

Proposition 2.17 *the relation $(Sim\beta) \cup (Sim\eta)$ is confluent on $\mathcal{D}$.*

By proposition 2.1, we get the following theorem:

Theorem 2.7 *The rewriting system $SL\beta\eta$ is confluent on $\mathcal{D}$.*

Technical Lemmas of Part B

Lemma 2.7 *Let A and B be two terms of $\mathcal{SD}$.*

1. *If $X \equiv App \circ < \Lambda(A), B >$ verifies $C(\eta)$, then $Y \equiv SL(A \circ < Id, B >)$ also verifies it.*
2. *If A verifies $C(\eta)$, if B belongs to $\mathcal{SD}$, then if $m \geq 1$, $A \circ \mathcal{P}^m(< Id, B >)$ also verifies it.*
3. *Let $M \in \mathcal{SD}$ be the context of a (Beta)-redex and N be its reduct:*

$$M \equiv C[\alpha \longleftarrow App \circ < \Lambda(A), B >] \text{ and } N \equiv SL(C[\alpha \longleftarrow SL(A \circ < Id, B >)])$$

If M verifies $C(\eta)$, then N also verifies it.

Proof
1–Let $(F; u)$ be a leaf of A. Let $p!$ be its extremity. This leaf is also one leaf of X. Its occurrence in X is $v \equiv 210u$. So we have:

$$p \neq (|v|, X) \text{ so } p \neq (|u|, A) + 1$$

let $(G; v)$ be one leaf of B .Let $q!$ be its extremity. We have:

$$q \neq (|22v|, X) \text{ so } q \neq (|v|, B)$$

From proposition 2.3 of the section 2.3, we get:

$$Y = A\,[F_i \longleftarrow SL(F_i \circ \mathcal{P}^{(|u_i|,A)}(B)\,)]$$

where the $(F_i; u_i)$ are the leaves of A.

We now examine the term $SL(F \circ \mathcal{P}^{(|u|,A)}(B)\,)$.

1. $p < (|u|, A)$
   Then $SL(F \circ \mathcal{P}^{(|u|,A)}(B))$ is a leaf of Y. Its occurrence in Y is still u. It verifies:

   $$SL(F \circ \mathcal{P}^{(|u_i|,A)}(B)) = F$$

   Now, we know that: $(|u|, A) = (|u|, X)$. So this leaf is not η-accessible in Y.

2. $p > (|u|, A)$
   Then $SL(F \circ \mathcal{P}^{(|u_i|,A)}(B))$ is still a leaf of Y. Its occurrence is u and its extremity is $(p-1)!$.
   Now:

   $$(|u|, A) = (|u|, X) \text{ and } p \neq (|u|, A) + 1$$

   So:

   $$p - 1 \neq p(|u|, X)$$

   This leaf is not η-accessible in X.

3. $p = (|u|, A)$
   Then:

$$SL(F \circ \mathcal{P}^{(|u_i|,A)}(B)) = SL(B \circ Fst^{(|u_i|,A)}) = B[G_j \longleftarrow G_j \circ \mathcal{P}^{(|v_j|,B)}(Fst^{(|u_i|,A)})]$$

   where G_j denotes the leaf at the occurrence v_j in B.

   We examine the following term where G is a leaf:

$$G' = SL(G \circ \mathcal{P}^{(|v|,B)}(Fst^{(|u|,A)}))$$

   (a) If $q < (|v|, B)$, then G' is a leaf. Its height in Y is $|u| + |v|$ and its extremity is $q!$. This leaf is not η-accessible in Y.
   (b) If $q > (|v|, B)$, then G' is still a leaf. Its height in Y is $|u| + |v|$ and its extremity is $(q + |u|)\,!$. This leaf is not η-accessible in Y.
   (c) by hypothesis $q \neq (|v|, B)$.

So none of Y's leaves is η-accessible.

2,3 –The second and third points are straightforward. Only remark that redex creations in a prefix of α in N does not modify occurrence's heights of remaining leaves in N. ■

Lemma 2.8 *let M be a term in $\mathcal{SD}$ containing a $(S\lambda_{SD})$-redex. Suppose that this redex itself contains a (Beta)-redex:*

$$M \equiv C_1[v \longleftarrow \Lambda(App \circ < R, Snd >)]$$

where

$$R \equiv C[\alpha \longleftarrow App \circ < \Lambda(A), B >]$$

let:

$$S \equiv SL(C[\alpha \longleftarrow A \circ < Id, B >]) \quad N \equiv C_1[v \longleftarrow \Lambda(App \circ < S, Snd >)]$$

$$P \equiv SL(C_1[v \longleftarrow R^{\downarrow}]) \quad Q \equiv SL(C_1[v \longleftarrow S^{\downarrow}])$$

Then:

$$M \overset{(Sim\beta)}{\longrightarrow} N \text{ and } M \overset{(Sim\eta)}{\longrightarrow} P$$

$$P \overset{(Sim\beta)^*}{\longrightarrow} Q \text{ and } N \overset{(Sim\eta)}{\longrightarrow} Q$$

Proof
Let $X = C_1[v \longleftarrow \Omega]$. As $X \in \mathcal{SD}$, Ω may only be the extremity of one leaf of X. Let p be this leaf's occurrence. We have only to examine the following sub-terms : $N|_p$, $P|_p$ and $Q|_p$. Let:

$$X|_p = k_1 \circ (k_2 \circ (\ldots k_n \circ \Omega) \ldots)$$

By hypothesis, no leaf in R is η-accessible. By lemma 2.7, S verifies also the condition $C(\eta)$. Therefore N still contains a $(S\lambda_{SD})$-redex . Q is obtained by reducing it:

$$Q = SL(C_1[v \longleftarrow S^{\downarrow}])$$

A (Beta)-redex may be present in P since:

$$P|_p = SL(k_1 \circ (k_2 \circ (\dots k_n \circ (C[\alpha \longleftarrow App \circ < \Lambda(A), B >])^{\downarrow})) =$$

$$SL(k_1 \circ (k_2 \circ (\dots k_n \circ (C^{\downarrow}[\alpha \longleftarrow App \circ < \Lambda(A^{\downarrow}), B^{\downarrow} >]\,])$$

where $C^{\downarrow}$ denotes the context C being decremented.
If this $(Beta)$-redex does not disappear by applications of a projection rule, then let P_1 be obtained by reducing it:

$$P_1|_p = SL(k_1 \circ (k_2 \circ (\dots k_n \circ (C^{\downarrow}[\alpha \longleftarrow A^{\downarrow} \circ < Id, B^{\downarrow} >]) \dots)$$

Now:

$$Q|_p = SL(k_1 \circ (k_2 \circ (\dots k_n \circ SL(C[\alpha \longleftarrow A \circ < Id, B >])^{\downarrow} >) \dots)$$

So it suffices to prove the following equality:

$$SL((C[\alpha \longleftarrow A \circ < Id, B >])^{\downarrow}) = SL(C^{\downarrow}[\alpha \longleftarrow A^{\downarrow} \circ < Id, B^{\downarrow} >])$$

which is made by a simple examination of the leaves. We use the following equality:

$$SL((A \circ < Id, B >)^{\downarrow}) = SL(A^{\downarrow} \circ < Id, B^{\downarrow} >) \qquad \blacksquare$$

Lemma 2.9 *let M be a term containing a (Beta)-redex and N be its $(Sim\beta)$-reduct:*

$$M \equiv C[\alpha \longleftarrow App \circ < \Lambda(A), B >] \text{ and } N = SL(C[\alpha \longleftarrow A \circ < Id, B >])$$

Suppose that A contains a $(S\Lambda_{SD})$-redex . Let P be the following $(Sim\eta)$-reduct of M:

$$A = C_1[u \longleftarrow \Lambda(App \circ < R, Snd >)\,] \quad P = C[\alpha \longleftarrow App \circ < \Lambda(SL(C_1[u \longleftarrow R^{\downarrow}])), B >]$$

Let:

$$Q = SL(C[\alpha \longleftarrow SL(C_1[u \longleftarrow R^{\downarrow}] \circ < Id, B >])\,]\,)$$

Then:

$$P \overset{(Sim\beta)}{\longrightarrow} Q \text{ and } N \overset{(Sim\eta)}{\longrightarrow} Q$$

Proof
let X be the term $C[\alpha \longleftarrow \Omega]$. As M is in SD, Ω may be only the extremity of a leaf $(F;p)$ of X. We have only to examine the terms $M|_p$, $N|_p$, $P|_p$ and $Q|_p$. Let:

$$X|_p = k_1 \circ (k_2 \dots \circ (k_n \circ \Omega) \dots)$$

So:

$$M|_p = k_1 \circ (k_2 \dots \circ (k_n \circ App \circ < \Lambda(A), B >) \dots)$$

$$P|_p = k_1 \circ (k_2 \dots \circ (k_n \circ App \circ < \Lambda(SL(C_1[u \longleftarrow R^{\downarrow}])), B >) \dots)$$

We apply the relation $(Sim\beta)$ to P at the occurrence α and we get the term Q such that:

$$Q|_p = SL(k_1 \circ (k_2 \dots \circ (k_n \circ (C_1[u \longleftarrow R^{\downarrow}] \circ < Id, B >)) \dots)$$

Now by an easy calculation we obtain:

$$Q|_p = SL(k_1 \circ (k_2 \ldots \circ (k_n \circ ($$
$$C_1[\quad u \longleftarrow R^{\downarrow} \circ \mathcal{P}^{(|u|,C_1)}(< Id, B >), \quad F_j \longleftarrow F_i \circ \mathcal{P}^{(|v_j|,C_1)+1}(< Id, B >)]) \ldots)$$

where the leaves F_j appearing in the above term are the leaves of A such that their occurrences v_j are strictly disjoint from u.

$$N|_p = SL(k_1 \circ (k_2 \ldots \circ (k_n \circ (A \circ < Id, B >) \ldots))$$

So:

$$N|_p = SL(k_1 \circ (k_2 \ldots \circ (k_n \circ C_1[u \longleftarrow \Lambda(App \circ < R, Snd >)] \circ < Id, B >) \ldots))$$

which is equal to:

$$N|_p = SL(k_1 \circ (k_2 \ldots \circ (k_n$$
$$\circ C_1[u \longleftarrow \Lambda(App \circ < R \circ \mathcal{P}^{(|u|,C_1)+1}(< Id, B >), Snd >),$$
$$F_j \longleftarrow F_i \circ \mathcal{P}^{(|v_j|,C_1)+1}(< Id, B >)]) \ldots))$$

where the occurrences v_j are all the occurrences of leaves F_j of A, which are strictly disjoint from u.
The $(S\Lambda_{SD})$-redex may disappear by putting N in SL-normal form. Let:

$$Z = SL(k_1 \circ (k_2 \ldots \circ (k_n \circ (C_1[u \longleftarrow \Omega]) \ldots)$$

There is at most one occurrence of Ω in Z since A is a term of $\mathcal{SD}$. If Z has no occurrence of Ω, then $N \equiv Q$. Else it suffices to prove the following equality:

$$SL(R \circ \mathcal{P}^{(|u|,C_1)+1}(< Id, B >))^{\downarrow} = SL(R^{\downarrow} \circ \mathcal{P}^{(|u|,C_1)}(< Id, B >))$$

to conclude that:

$$N \overset{(Sim\beta)}{\longrightarrow} Q \blacksquare$$

Lemma 2.10 *let M be a term containing a (Beta)-redex and N be its $(Sim\beta)$-reduct:*

$$M \equiv C[\alpha \longleftarrow App \circ < \Lambda(A), B >] \text{ and } N = SL(C[\alpha \longleftarrow A \circ < Id, B >])$$

Suppose that B contains a $(S\lambda_{SD})$-redex and that P is the following $(Sim\eta)$-reduct of M:

$$B = C_1[u \longleftarrow \Lambda(App \circ < R, Snd >)] \quad P = C[\alpha \longleftarrow App \circ < \Lambda(A), SL(C_1[u \longleftarrow R^{\downarrow}]) >]$$

Let:

$$Q = SL(C[\alpha \longleftarrow SL(A \circ < Id, C_1[u \longleftarrow R^{\downarrow}] >)])$$

Then:

$$P \overset{(Sim\beta)}{\longrightarrow} Q \text{ and } N \overset{(Sim\eta)^*}{\longrightarrow} Q$$

Proof
Use proposition 2.3 to obtain to obtain the following equalities:

$$N = SL(C[\alpha \longleftarrow A[F_i \longleftarrow F_i \circ \mathcal{P}^{(|u_i|,A)}(< Id, C_1[u \longleftarrow \Lambda(App \circ < R, Snd >)] >)]])$$
$$Q = SL(C[\alpha \longleftarrow A[F_i \longleftarrow F_i \circ \mathcal{P}^{(|u_i|,A)}(< Id, SL(C_1[u \longleftarrow R^{\downarrow}]) >)]])$$

3 $(\mathcal{D}, SL\beta\eta)$: a confluent conservative extension of λ-calculi

We have now to relate the different λ-calculi and this subset $\mathcal{D}$ of C.C.L. In this section, we describe a bijection between Λ and a sub-set $\mathcal{SD}_\lambda$ de $\mathcal{SD}$ which is extended as a bijection between $\beta\eta$-derivations of Λ and derivations of $\mathcal{SD}_\lambda$ by the relation $(Sim\beta)\cup(Sim\eta)$. Then these results are extended to $(\Lambda_{c,f}\ ,\ \beta\eta P)$.

3.1 $\mathcal{D}$ and Λ

We rewrite the translation $(\ )_{DB\{x_o,\ldots,x_n\}}$ from $\Lambda(V)$ into C.C.L into a translation between Λ and C.C.L.

Definition 3.1 Let $M \in \Lambda$. The **translation** M_D of M into C.C.L. is inductively defined by :

1. If $M = n$, then $M_D = Snd \circ Fst^n$ (denoted by n!)
2. If $M = N\ P$, then $M_D = App \circ < N_D, P_D >$ (denoted by $\theta(N_D, P_D)$)
3. If $M = \lambda(N)$, then $M_D = \Lambda(N_D)$

The subset $\mathcal{SD}_\lambda$ is defined by the following equality :

$$\mathcal{SD}_\lambda\ = \{M_D | M \in \Lambda\}$$

Let $P \in \mathcal{SD}_\lambda$. The **translation** P_λ of P into Λ is defined inductively by :

1. If $P = n!$, then $P_\lambda = n$
2. If $P = App \circ < S, T >$, then $P_\lambda = S_\lambda\ T_\lambda$
3. If $P = \Lambda(N)$, then $P_\lambda = \lambda(N_\lambda)$

The subset $\mathcal{D}_\lambda$ of $\mathcal{D}$ is defined by the following equation :

$$\mathcal{D}_\lambda = (SL\beta\eta)^*(\mathcal{SD}_\lambda)$$

Proposition 3.1 *$(Sim\beta)$ and $(Sim\eta)$ are internal relations of $\mathcal{SD}_\lambda$. If $P \in \mathcal{D}_\lambda$, then $SL(P) \in \mathcal{SD}_\lambda$*

Proof
The first part is straightforward. The second part is as follows. By definition there exists $M \in \mathcal{SD}_\lambda$ such that $P = (SL\beta\eta)^n(M)$. Using proposition 2.3, we obtain :

$$M\ \ (Sim\beta)^* \cup (Sim\eta)^*\ \ SL(P)$$

Now we have a bijection between Λ and $\mathcal{SD}_\lambda$:

Proposition 3.2

$$\forall M \in \Lambda\ ,\ \ M_D \in \mathcal{SD}_\lambda\ \text{ and }\ (M_D)_\lambda \equiv M$$

$$\forall P \in \mathcal{SD}_\lambda\ ,\ \ P_\lambda \in \Lambda\ \text{ and }\ (P_\lambda)_D \equiv P$$

where $\equiv$ is the identity.

The *position* of a subterm P in the term M of $\mathcal{SD}_\lambda$ is by definition the occurrence of P as a subterm of M written on the alphabet $\{\theta\ ,\ \Lambda\ ,\ n!\}$.

We extend the previous bijection to derivations :

Theorem 3.1 *Let M and N be two terms of Λ.*

$$\text{If } M \overset{\beta^\star \cup \eta^\star}{\longrightarrow} N \quad \text{, then } M_D \overset{(Sim\beta)^\star \cup (Sim\eta)^\star}{\longrightarrow} N_D$$

Let P and Q be two terms of $\mathcal{SD}_\lambda$.

$$\text{If } P \overset{(Sim\beta)^\star \cup (Sim\eta)^\star}{\longrightarrow} Q \quad \text{, then } P_\lambda \overset{\beta^\star \cup \eta^\star}{\longrightarrow} Q_\lambda$$

Moreover an occurrence is the one of a β-redex (resp. of a η-redex) if and only if the coresponding position is the one of a (Beta)-redex (resp. of a $(Sim\eta)$-redex).

Proof

Let $P = C[u \longleftarrow App \circ < \Lambda(A), B >]$ and $Q = C[u \longleftarrow SL(A \circ < Id, B >)\ ]$. We have :

$$P_\lambda = C_\lambda[u \longleftarrow \lambda(A_\lambda)B_\lambda)\ ]$$

Let Q' be obtained by reducing the β-redex at the occurrence u of P_λ:

$$Q' = C_\lambda[u \longleftarrow \sigma_0(A_\lambda, B_\lambda)\ ]$$

To get $Q' \equiv Q_\lambda$, we only have to prove :

$$\sigma_0(A_\lambda, B_\lambda) \ \equiv (SL(A \circ < Id, B >))_\lambda$$

We prove for any n, A and $B \in \mathcal{D}_\lambda$, the following equality :

$$\sigma_n(A_\lambda, B_\lambda) \ = (SL(A \circ \mathcal{P}^n(< Id, B >)))_\lambda$$

by induction on $SL(A)$. The only non trivial case is $A = m!$. We get:

$$(SL(m! \circ \mathcal{P}^n(< Id, B >)))_\lambda = \begin{array}{lll} \text{if} & m > n & (m-1) \\ \text{if} & m < n & m \\ \text{if} & m = n & (SL(B \circ Fst^n))_\lambda \end{array}$$

It remains to prove the following equality :

$$\forall n \in \mathrm{N}, \forall B \in \mathcal{D}_\lambda\ ,\ \ \tau_i^n(B_\lambda) = (SL(B \circ \mathcal{P}^i(Fst^n)))_\lambda$$

This is done by induction on $SL(B)$, for all n.

The converse result and the one for η-reduction are obtained in the same way. Note only that the height in λ of an occurrence in a λ-term is the same as the height of the corresponding position in its translation. ■

Theorem 3.2 *($\mathcal{D}$,$SL\beta\eta$) is a conservative confluent extension of (Λ , $\beta\eta$).*

Proof

We have only to collect the previous results. $(\ )_D$ is an injection of Λ into $\mathcal{D}$, such that the two points required in the definition 1.2 are fulfilled. ■

Note that the relations $(Sim\beta)$ and $(Sim\eta)$ reproduce exactly the relations β and η. So all the classical results of λ-calculus theory – Finite Developpements, Standardisation, Normalisation ... – may be carried into C.C.L. as properties of the relations $(Sim\beta)$ and $(Sim\eta)$.

3.2 $\Lambda_{\mathbf{c,f}}$ and C.C.L.

We now add to Λ the coupling operator which will be translated into C.C.L. by the pairing operator.

Definition 3.2 Let $M \in \Lambda_{\mathbf{c,f}}$. The translation M_D of M is defined as an extension of the translation M_D of Λ by adding the following points :

1. If $M =< N, P >$, then $M_D =< N_D, P_D >$
2. If $M = fst(N)$, then $M_D = Fst \circ N_D$
3. If $M = snd(N)$, then $M_D = Snd \circ N_D$

Remark 3.1 If $M \in \Lambda_{\mathbf{c,f}}$, M_D is not in general a term in SL-normal form, due to the possible projection redexes of M. Here we have two possibilities: either we add labellings to *Fst* and *Snd* to study the correspondance between $\Lambda_{\mathbf{c,f}}$ and C.C.L. or we use the c-normal form of M, that is the normal form of M for the rewriting system ((Fst),(Snd)). We first examine this point.

Definition 3.3 let $\mathcal{SD}_{P\lambda}$ be the subset of $\mathcal{SD}$ defined by the following equality :

$$\mathcal{SD}_{P\lambda} = \{c(M)_D | M \in \Lambda_{\mathbf{c,f}}\}$$

Let $P \in \mathcal{SD}_{P\lambda}$. The translation P_λ of P into $\Lambda_{\mathbf{c,f}}$ is an extension of the translation P_λ of $\mathcal{SD}_\lambda$ obtained by adding the following points :

1. If $P =< S, T >$, then $P_\lambda =< S_\lambda, T_\lambda >$
2. If $P = Fst \circ N$, then $P_\lambda = fst(N_\lambda)$
3. If $P = Snd \circ N$, then $P_\lambda = snd(N_\lambda)$

$\mathcal{D}_{\mathcal{P}\lambda}$ is the subset of $\mathcal{D}$ defined by :

$$\mathcal{D}_{\mathcal{P}\lambda} = (SL\beta\eta)^\star(\mathcal{SD}_{P\lambda})$$

Proposition 3.3 *The relations $(Sim\beta)$ and $(Sim\eta)$ are internal to $\mathcal{SD}_{P\lambda}$. If $P \in \mathcal{D}_{\mathcal{P}\lambda}$, then $SL(P) \in \mathcal{SD}_{P\lambda}$*

Proposition 3.4

$$\forall M \in \Lambda_{\mathbf{c,f}}\ ,\ c(M)_D \in \mathcal{SD}_{P\lambda} \text{ and } (c(M)_D)_\lambda \equiv c(M)$$

$$\forall P \in \mathcal{SD}_\lambda\ ,\ P_\lambda \in \Lambda_{\mathbf{c,f}} \text{ and } (P_\lambda)_D \equiv P$$

Theorem 3.3 *Let M and N be two terms of $\Lambda_{\mathbf{c,f}}$.*

$$\text{If } M \overset{\beta^\star\cup\eta^\star}{\longrightarrow} N \quad \text{, then } c(M)_D \overset{(Sim\beta)^\star\cup(Sim\eta)^\star}{\longrightarrow} c(N)_D$$

Let P and Q be two terms of $\mathcal{SD}_{P\lambda}$.

$$\text{If } P \overset{(Sim\beta)^\star\cup(Sim\eta)^\star}{\longrightarrow} Q \quad \text{, then } P_\lambda \overset{\beta^\star\cup\eta^\star}{\longrightarrow} Q_\lambda$$

Moreover an occurrence is the one of a β-redex (resp. of a η-redex) if and only if the coresponding position is the one of a (Beta)-redex (resp a $(Sim\eta)$-redex).

Now we become precise about labellings. We add to C.C.L. two constants Fst_1 and Snd_1 and the corresponding rules. We may reproduce the previous work: $\mathcal{E}$ is still a confluent system on C.C.L$_1$.

Let $M \in \mathcal{D}_1$ iff $\mathcal{E}(M)$ as only well-formed leaves. Then we can describe easily this term M: replacing Fst_1 and Snd_1 by App in M must lead to a term in $\mathcal{D}$. Let $SL\beta\eta_0$ be the rewriting system defined by the rules of $SL\beta\eta$ on C.C.L.$_1$.

The $\mathcal{E}$-interpretation of $SL\beta\eta_0$ is now a relation $(Sim\beta)_0 \cup (Sim\eta)_0$ defined like $(Sim\beta) \cup (Sim\eta)$. It does not reduce (Fst_1) and (Snd_1) redexes. Moreover the $\mathcal{E}$-interpretation of the rewriting relation defined by $((Fst)_1 \cup (Snd)_1)^\star$ is itself: this point is easily proved by examining the redex creations during the sticking of the fragments.

Let $SL\beta\eta_1$ be $SL\beta\eta_0 \cup (Fst)_1 \cup (Snd)_1$. Its $\mathcal{E}$-interpretation, called $Sim\beta\eta P$ is the union of $((Sim\beta)_0 \cup (Sim\eta)_0)^\star$ and $((Fst)_1 \cup (Snd)_1)^\star$.

Definition 3.4 The **translation** $M_{D,1}$ from $\Lambda_{\mathbf{c,f}}$ into $\mathcal{D}_1$ is defined by replacing in the definition of M_D Fst by Fst_1 and Snd by Snd_1 in the points 2 and 3.
let $\mathcal{SD}_{P\lambda 1}$ be the subset of $\mathcal{SD}_1$ defined by the following equality :

$$\mathcal{SD}_{P\lambda 1} = \{M_{D,1} | M \in \Lambda_{\mathbf{c,f}}\}$$

Let $P \in \mathcal{SD}_{P\lambda 1}$. The **translation** $P_{\lambda,1}$ of P into $\Lambda_{\mathbf{c,f}}$ is also an extension of the translation P_λ of $\mathcal{SD}_\lambda$ obtained by adding the following points :

1. If $P = < S, T >$, then $P_\lambda = < S_\lambda, T_\lambda >$
2. If $P = Fst_1 \circ N$, then $P_\lambda = fst(N_\lambda)$
3. If $P = Snd_1 \circ N$, then $P_\lambda = snd(N_\lambda)$

$\mathcal{D}_{\mathcal{P}\lambda 1}$ is the subset of $\mathcal{D}_1$ defined by :

$$\mathcal{D}_{\mathcal{P}\lambda 1} = (SL\beta\eta_1)^\star(\mathcal{SD}_{P\lambda 1})$$

The proofs of the following propositions are identical as the corresponding ones in the previous section.

Proposition 3.5 *The relation $Sim\beta\eta P$ is internal to $\mathcal{SD}_{P\lambda 1}$. If $P \in \mathcal{D}_{\mathcal{P}\lambda 1}$, then $\mathcal{E}(P) \in \mathcal{SD}_{P\lambda 1}$.*

Proposition 3.6

$$\forall M \in \Lambda_{\mathbf{c,f}}, \ M_{D,1} \in \mathcal{SD}_{P\lambda 1} \text{ and } (M_{D,1})_{\lambda,1} \equiv M$$

$$\forall P \in \mathcal{SD}_{P\lambda 1}, \ P_{\lambda,1} \in \Lambda_{\mathbf{c,f}} \text{ and } (P_{\lambda,1})_{D,1} \equiv P$$

Theorem 3.4 *Let M and N be two terms of $\Lambda_{\mathbf{c,f}}$.*

$$\text{If } M \stackrel{\beta\eta P^\star}{\longrightarrow} N \text{ , then } M_{D,1} \stackrel{Sim\beta\eta P}{\longrightarrow} N_{D,1}$$

Let P and Q be two terms of $\mathcal{SD}_{P\lambda 1}$.

$$\text{If } P \stackrel{(Sim\beta\eta P)^\star}{\longrightarrow} Q \text{ , then } P_{\lambda,1} \stackrel{\beta\eta P^\star}{\longrightarrow} Q_{\lambda,1}$$

Remark 3.2 Suppose that $P_1 \in \mathcal{D}_{P\lambda 1}$. Let P be obtained by erasing the labels in P_1. Suppose that Q is derived from P by $SL\beta\eta$ in C.C.L. Then there exists a labelled derivation of P_1 leading to Q_1 such that Q is obtained from Q_1 by erasing the labels: we have only to label the applications of projection rules when the involved projection is labelled.

Theorem 3.5 *$(\mathcal{D}, SL\beta\eta)$ is a confluent conservative extension of $(\mathbf{\Lambda_{c,f}}, \beta\eta P)$.*

Proof

First, we prove that the point 1 of the definition 1.2 is fulfilled.
Let M a $N \in \mathbf{\Lambda_{c,f}}$. $(\,)_D$, which is $(\,)_{D,1}$ followed by an erasing of labels, is clearly an injection from $\mathbf{\Lambda_{c,f}}$ into $\mathcal{D}$. Erasing the labels on terms and rules leads to $SL\beta\eta$-derivations of C.C.L. so if $M\ \beta\eta P\ N$ in $\mathbf{\Lambda_{c,f}}$ then $M_D\ SL\beta\eta\ N_D$. Conversely, suppose that $M_D\ SL\beta\eta\ N_D$. Then $M_{D,1}$ is a labelling of M_D and we can construct a labelled derivation from $M_{D,1}$ to $N_{D,1}$ such that $N_{D,1}$ is a labelling of N_D. Moreover $N_{D,1} \in \mathcal{D}_{P\lambda 1}$ therefore in $\mathcal{SD}_{P\lambda 1}$. Then $N_{D,1}$ is also a $(Sim\beta\eta P)$-derived of $M_{D,1}$. By using theorem 3.4 we get:$M\ \beta\eta P\ N$.

Now we get the second point of definition 1.2.
Let $M \in \mathbf{\Lambda_{c,f}}$ and suppose that $M_D\ SL\beta\eta\ Q$. Then the labelling $M_{D,1}$ provides a labelling Q_1 of Q such that $M_{D,1}\ SL\beta\eta_1\ Q_1$. Then, $\mathcal{E}(Q_1) \in \mathcal{SD}_{P\lambda 1}$. ∎

4 $CCL\beta SP$ is not confluent

4.1 Yet another counter-example for $\mathbf{\Lambda_{c,a}}$

Our counter-example is an improvement from Klop's one. We construct a term B which has a normal form I and a reduct CI. By a simple examination of derivations of CI eventually leading to I, we prove that I cannot be a reduct of CI.

Notations

Let $P = \lambda x \lambda y.y((xx)y)$. Let $Y_T = P\,P$ be the Turing fixed point.
Let $U = \lambda x\lambda y.D(F(\lambda z.z(xy))(S(\lambda z.zy))(\lambda z.I)$ where $I = \lambda x.x$.
Let $C = Y_T\,U$ and $B = Y_T\,C$

Lemma 4.1 *I and $C\,I$ are two reducts of B.*

Proof

The typical derivations are:

$$C \xrightarrow{\beta^*} UC \qquad B \xrightarrow{\beta^*} C\,B$$

For any M in $\mathbf{\Lambda_{c,f}}$,

$$CM \xrightarrow{\beta^*} D(F(\lambda z.z\,(C\,M)))(S(\lambda z.z\,M))(\lambda z.I) \equiv X$$

Therefore
$B \xrightarrow{\beta^*} C\,B \xrightarrow{\beta^*} D(F(\lambda z.z\,(C\,B)))(S(\lambda z.z\,B))(\lambda z.I)$
$\xrightarrow{\beta^*} D(F(\lambda z.z\,(C\,B)))(S(\lambda z.z\,(C\,B)))(\lambda z.I)$
$\xrightarrow{(SP)} (\lambda z.z\,(C\,B))(\lambda z.I) \xrightarrow{\beta} (\lambda z.I)(C\,B) \xrightarrow{\beta} I$
and:
$B \xrightarrow{\beta^*} C\,(C\,B) \xrightarrow{D_1} C\,I$ ∎

Lemma 4.2 *Let $M \in \mathbf{\Lambda}_{\mathbf{c},\mathbf{a}}$ having a normal form distinct from I.*
If βSP (resp. $\beta\eta SP$) verifies the uniqueness property for normal forms then $C\,M$ and M have no common reduct.

Proof
The term X of the previous lemma is a reduct of $C\,M$. If M and $C\,M$ have a common reduct A, then X can be rewritten on:

$$D(F\ (\lambda z.zA\,))(S\ (\lambda z.zA\,)\,)\ (\lambda z.I)$$

and then on I. ∎

Definition 4.1 A rewriting system R verifies the Property (NF) if:
let M be a term in R-normal form. Let N be equal to M. Then M can be obtained from N by an R-derivation.

Proposition 4.1 *The theories ($\mathbf{\Lambda}_{\mathbf{c},\mathbf{a}}$,$\beta SP$) and ($\mathbf{\Lambda}_{\mathbf{c},\mathbf{a}}$, $\beta\eta SP$) are not confluent: the propoerty (NF) is not verified.*

Proof
We prove that CI cannot be reduced on I. We do it for β-derivations. The same holds for $\beta\eta$-derivations.
Let (R) be a derivation from $C\,I$ to I. CI contains only one redex: the one in Y_T. So (R) begins by performing the following step of reduction leading to the term A_1:

$$A_1 = (\,(\,\lambda y.y\,(Y_T\,y)\,)\,U\,)\,I$$

(R) may go on by deriving the sub-term $Y_T\,y$. But necessarily (R) should perform the leftmost-redex's reduction in order to reach I. So (R) contains a term A_2:

$$A_2 = U(Red(Y_T\,y)[y \longleftarrow\ U]\,)\,I = U(Red(Y_T U)\,)\,I = U(Red(C))\,I$$

where the notation $Red(X)$ indicates a reduct of X or the term X.
R may continue by deriving the sub-term $Red(C)$ but necessarily should reduce the leftmost-redex (defined by the top- λ of U). So R contains a term A_3:

$$A_3 \equiv (\lambda y.(D\ (F\ (\lambda z.z(Red\,(C)y)))\ (S\ (\lambda z.zy))\,)\ (\lambda z.I)\,)\ I$$

There is a sub-term $D(F..)(S..)$ in A_3. This context can only disappear by a (SP)-reduction. If (R) contains such a step before the reduction of the top-redex, then (R) contains one derivation from Cy to y. It is impossible (lemma 4.2) since, if βSP is confluent, then it has the uniqueness property for normal forms. So, before removing this context, (R) has to reduce the top-redex: (R) contains a term A_4:

$$A_4 \equiv D(\,(F\ (\lambda z.zRed(CI))\,)\ (S\ (\lambda z.zI)\,))\ (\lambda z.I)$$

We define the length of a derivation as the the number of (SP)-steps it contains. Let $Rmin$ a derivation from CI to I of minimal lenght. $Rmin$ has to remove the context $DF(..)S(..)(\lambda z.I)$. So it contains a derivation from CI to I and is not of minimal length. ∎
To obtain a counter-example for $\mathbf{\Lambda}_{\mathbf{c},\mathbf{f}}$, it suffices to replace in the precedent proof the term U by the term W:

$$W \equiv \lambda x\,\lambda y(<\ fst\,(\lambda z.\ z(xy)\,)\ ,\ snd(\,\lambda z.zy\,)\ >\ (\lambda z.\,I)\,)$$

4.2 The relation βSP is not confluent

Definition 4.2

$$\beta SP \ = (Sim\beta) \cup (SP) \quad \beta\eta SP = \beta SP \cup (Sim\eta)$$

This definition is correct since the rewriting relation (SP) is proved to be internal to $\mathcal{SD}$: reducing a (SP)-redex in a term of $\mathcal{SD}$ gives a reduct still in $\mathcal{SD}$.

We use the translation M_D to reproduce the counter-example for $\Lambda_{c,f}$ into C.C.L. The derivations become an iteration of the relation βSP so either a $(Sim\beta)$-step or a (SP)-step. The translations keep the name of the corresponding terms:

$$\begin{array}{rcl} P & = & \Lambda\Lambda(\theta(0!, \theta(\theta(1!, 1!), 0!))) \\ Y_T & = & \theta(P, P) \\ I & = & \Lambda(0!) \\ U & = & \Lambda\Lambda(\theta(< Fst \circ \Lambda(\theta(0!, \theta(2!, 1!))) \\ & & \qquad\qquad\qquad , Snd \circ \Lambda(\theta(0!, 1!)) >, \Lambda(I))) \\ C & = & \theta(Y_T, U) \\ B & = & \theta(Y_T, C) \end{array}$$

Remark 4.1 All these terms belong to $\mathcal{SD}$.

Needed lemmas are as follows:

Lemma 4.3 1. For any term M of C.C.L., the following holds:

$$\theta(Y_T, M) \ \beta SP^\star \ \theta(M, \theta(Y_T, M))$$

2. We get: $B \ \beta SP^\star \ \theta(C, B)$
3. so:

$$\theta(C, M) \ \beta SP^\star \theta(< Fst \circ \Lambda(\theta(0!, \theta(C, SL(M \circ Fst)))), Snd \circ \Lambda(\theta(0!, SL(M \circ Fst))) > , \ \Lambda(I))$$

Lemma 4.4 If βSP is confluent, then I is the $CCL\beta SP$-normal form of $\theta(C, I)$.

Follows the lemma corresponding to lemma 4.2. Note that the hypothesis on M is replaced by the same hypothesis on $M \circ Fst$, due to the previous lemma.

Lemma 4.5 If βSP is confluent, if there exists a common reduct of the two terms $SL(M \circ Fst)$ and $\theta(C, SL(M \circ Fst))$ then I is the βSP-normal form of $SL(M \circ Fst)$.

Lemma 4.6 Substitution Lemma for βSP
Let $M \in \mathcal{SD}$. If $M \ \beta SP \ N$, then:

$$SL(M \circ \mathcal{P}^m(< Id, U >)) \ \beta SP^\star \ SL(N \circ \mathcal{P}^m(< Id, U >))$$

Note that this lemma is no longer verified if we replace the relation βSP by the relation which consists in firstly reducing a (Beta)-redex and then putting the reduct in $Subst$-normal form.

Theorem 4.1 *The relations βSP and $\beta\eta SP$ are not confluent.*

Proof
The method is the one of $\Lambda_{c,f}$. We give it as an example. There is only one redex in $\theta(C, I)$: the one of $\theta(P, P)$. Its reduction gives the following term A_1:

$$A_1 = \theta(\theta(\Lambda(\theta(0!, \theta(Y_T, 0!))), U), I)$$

Let $X_1 = \theta(Y_T, 0!)$. We get:

$$A_1 = \theta(\theta(\Lambda(\theta(0!, X_1)), U), I)$$

X_1 contains redexes but in order to get I, the top-redex must be reduced so we get the following term:

$$A_2 = \theta(\theta(U, SL(\beta SP^\star(X_1) \circ < Id, U >)), I)$$

By lemma 4.6:

$$SL(X_1 \circ < Id, U >) \ \beta SP^\star \ SL(\beta SP^\star(X_1) \circ < Id, U >)$$

So:

$$A_2 = \theta(\ \theta(U, \beta SP^\star(\theta(Y_T, U))\ )\ ,\ I)$$

and:

$$A_2 = \theta(\ \theta(U, \beta SP^\star(C))\ ,\ I)$$

Let $X_2 = \beta SP^\star(C)$. This subterm contains redexes but the following top-redex must be reduced:

$$\theta(U, X_2)$$

Reducing it leads to the following term X_3:

$$X_3 = \Lambda(\theta(< Fst \circ \Lambda(\theta(0!, \theta(X_2, 1!))), Snd \circ \Lambda(\theta(0!, 1!)) >\ ,\ \Lambda(I)\ )\ )$$

so, to the following term A_3:

$$A_3 = \theta(X_3, I)$$

This term X_3 contains the following sub-term $< Fst \circ -, Snd \circ - >$. If βSP is confluent then this sub-term cannot disappear by an application of (SP): by lemma 4.5, $1!$ is not a reduct of $\theta(C, 1!)$ since I is not equal to $1!$. Therefore any derivation must reduce the top-redex of A_3 so contains the following term A_4:

$$A_4 = \theta(< Fst \circ \Lambda(\theta(0!, X_4)), Snd \circ \Lambda(\theta(0!, I)) >,\ \Lambda(I))$$

where:

$$X_4 = SL(\beta SP^\star(\theta(C, 1!)) \circ \mathcal{P}^1(< Id, I >))$$

By lemma 4.6, we get the following:

$$X_4 = \beta SP^\star(\theta(C, I))$$

and, we obtain the following equality:

$$A_4 = \theta(< Fst \circ \Lambda(\ \theta(0!, \beta SP^\star(\theta(C, I))\ )\ ), Snd \circ \Lambda(\theta(0!, I)) >\ ,\ \Lambda(I)\ )$$

We conclude with the method of $\Lambda_{c,f}$. ∎

Remark 4.2 $(Sim\beta) \cup (Sim\eta) \cup (SP)$ also is not confluent.

4.3 *CCLβSP* is not confluent

CCLβSP is a weakly confluent system. We show that this relation βSP is the $\mathcal{E}$-interpretation of the rewriting relation of *CCLβSP* on $\mathcal{D}$ so we can prove that this system is not confluent. But when restricted to $\mathcal{D}_\lambda$ this system *CCLβSP* is indeed confluent. Moreover the non-weakly confluent system *CCLβηSP* is confluent on $\mathcal{D}_\lambda$.

First we interpret the rewriting relation (SP) ∪ (FSI) on $\mathcal{D}$.

Proposition 4.2 *Let* $M \in \mathcal{D}$.

$$\text{if } M \overset{(SP)\cup(FSI)}{\longrightarrow} N \text{ then } SL(M) \overset{(SP)^*}{\longrightarrow} SL(N)$$

Proof
Let $P = C[u \longleftarrow \Omega]$ and $SL(P) = C'[a_p \longleftarrow \Omega \circ Q_p\,;\, b_q \longleftarrow \Omega]$.
For all i, if $i = p$, then X_i denotes $SL(X \circ Q_p)$ and $u_i = a_p$; if $i = q$ then X_i denotes $SL(X)$ and $u_i = b_q$. Now the following term is a reduct of M.

$$M_1 = C'[u_i \longleftarrow\ < SL(Fst \circ X_i), SL(Snd \circ X_i) >]$$

Let α_i be the father of u_i. There are two cases:

1. α_i is not an occurrence of a redex in M_1. The pairing at the occurrence u_i cannot disappear. The term X_i has two possible forms:

   (a) $X_i =< X_{i1}, X_{i2} >$. Let v_j be such an occurrence u_i. We have:
   $< SL(Fst \circ X_i), SL(Snd \circ X_i) >=< X_{i1}, X_{i2} >$

   (b) $< SL(Fst \circ X_i), SL(Snd \circ X_i) >=< Fst \circ X_{i1}, Snd \circ X_{i2} >$
   Let γ_l be such an occurrence u_i.

   The other occurrences u_i will be indexed by k.

2. Sticking up the fragment into the interpretation of the context can create redexes:

$$M_2|_{\alpha_k} = Fst \circ < SL(Fst \circ X_k), SL(Snd \circ X_k) >$$

   or

$$M_2|_{\alpha_k} = Snd \circ < SL(Fst \circ X_k), SL(Snd \circ X_k) >$$

   (We suppose that this created redex is always a (Fst)-redex)
   The pairing at occurrence u_k disappears by reduction of this redex and there is no redex creation "under" the buffer occurrence. Let w_k be the buffer occurrence of u_k.

By lemma 2.8, we get:

$$SL(M) = C'[\begin{array}{lll} \gamma_l & \longleftarrow & < Fst \circ X_l, Snd \circ X_l > \\ v_j & \longleftarrow & X_j \\ w_k & \longleftarrow & SL(C'|_{w_k}[\alpha_k \longleftarrow Fst \circ X_k])] \end{array}$$

From $SL(M) \in \mathcal{D}$, we deduce $Fst \circ X_l$ and $Snd \circ X_l \in \mathcal{D}$. From these hypothesis we prove by induction that $X_l \in \mathcal{D}$. Moreover reducing (SP)-redexes cannot create (SL)-redexes in a prefix occurrence of γ_l since ($X_l \neq Id$).

From a derivation from P to $SL(P)$, we build one from N to N_1:

$$\begin{array}{rll} N_1 = C'[& \gamma_l & \longleftarrow X_l \\ & v_j & \longleftarrow X_j \\ & w_k & \longleftarrow SL(C'|_{w_k}[\alpha_k \longleftarrow Fst \circ X_k])] \end{array}$$

Now $N_1 = SL(N)$ so we get:

$$SL(M) \stackrel{(SP)^*}{\longrightarrow} SL(N) \text{ and } SL(N) \in \mathcal{D} \blacksquare$$

Now we give the interpretations of $CCL\beta SP$ and of $CCL\beta\eta SP$: they are βSP and $\beta\eta SP$.

Theorem 4.2 *Let $M \in \mathcal{D}$. Then:*

$$\begin{array}{llllll} \text{If } M & CCL\beta SP & N & \text{then} & SL(M) \; \beta SP^* \; SL(N) \\ \text{If } M & CCL\beta\eta SP & N & \text{then} & SL(M) \; \beta\eta SP^* \; SL(N) \end{array}$$

Proof
We only have to construct the following diagram:

$$\begin{tikzcd} M \arrow[r, "Subst"] \arrow[d] & M_1 \arrow[r, "(Beta)"] \arrow[d] & M_2 \arrow[r, "(AI)\cup(S\Lambda)"] \arrow[d] & M_3 \arrow[d] \\ SL(M) \arrow[r, "(SP)^*"] & SL(M_1) \arrow[r, "Sim\beta^*"] & SL(M_2) \arrow[r, "Sim\eta^*"] & SL(M_3 \end{tikzcd}$$

Theorem 4.3 *$CCL\beta SP$ is not a confluent system.*

Proof
Use proposition 2.1.

Now these negative results do not restrict calculations on terms translated from λ-calculus: the systems $CCL\beta SP$ and $CCL\beta\eta SP$ are confluent on $\mathcal{D}_\lambda$.

Theorem 4.4 *$CCL\beta SP$ and $CCL\beta\eta SP$ are confluent on $\mathcal{D}_\lambda$.*

Proof
Examine the diagram of theorem 4.2. If $M \in \mathcal{D}_\lambda$ then $SL(M) \in \mathcal{SD}_\lambda$ and contains no (SP)-redex. So the induced steps $(SP)^*$ on $\mathcal{SD}_\lambda$ are only identity steps and the $\mathcal{E}$-interpretation of $CCL\beta\eta SP$ ($CCL\beta SP$) on $\mathcal{D}_\lambda$ is the relation $(Sim\beta)^* \cup (Sim\eta)^*$ ($(Sim\beta)^*$). Therefore $CCL\beta\eta SP$ ($CCL\beta SP$) is confluent on $\mathcal{D}_\lambda$. $\blacksquare$

Remark 4.3 From a term in $\mathcal{D}_\lambda$ we can derive a term containing a (SP)-redex. It is due to the "implicit" representation of the environnement by t-uples. Such instances of (SP)-redexes can be reduced safely: this is the previous theorem. So non-confluence is explicitely due to the construction of the couple of two terms by the aim of the pairing operator. Moreover, note that a term of $\mathcal{SD}_{\lambda 1}$ does not contain non-labelled (SP)-redexes: $CCL\beta\eta SP$ remains confluent on C.C.L.$_1$. Non-confluence arrives with the labelled Surjective Pairing Rule.

Theorem 4.5 *$CCL\beta\eta SP$ is a conservative extension of ($\Lambda_{c,f}$, $\beta\eta SP$).*

proof

$(\,)_D$ is an injection of $\Lambda_{c,f}$ in C.C.L. such that the two points of definition 1.2 are easily verified, with $CCL\beta\eta SP$. We have to prove that this extension is conservative.

Let $\Lambda(V)_c$ be the λ-calculus with explicit couples, defined on a set V of variables. A denumbrable list of variables $x_1, \ldots, x_n \ldots$ being given, there exists an isomorphism, denoted by M^1 for $M \in \Lambda_{c,f}$, between $\Lambda(V)_c$ and $\Lambda_{c,f}$ such that $M_D \equiv M^1_{DB(x_0,\ldots,x_n)}$ (see M.Mauny 's thesis [17]).

Let M and $N \in \Lambda_{c,f}$ and suppose that $M_D =_{CCL\beta\eta SP} N_D$. We have:
$M^1_{DB(x_0,\ldots,x_n)} = N^1_{DB(x_0,\ldots,x_n)}$ therefore: $M^1_{CCL} = N^1_{CCL}$.
Then by using Curien Equivalence Theorem, we get the following:
$M^1 =_{\beta\eta SP} N^1$ and then $M =_{\beta\eta SP} N$. ■

Conclusion

Combinators have been widely studied since Schönfinkel and Curry's results: among them one may notice the works of Hindley, Scott, Meyer, Lambek, Koymans, Curien, Poigné which developed both the semantical aspects and the syntactical points of vue.

This work proves that Strong Categorical Logic is the good language to be chosen as an intermediate between machine languages and high level languages. We may reproduce not only the weak β-reduction (as it is done in the Classical Combinatory Logic) but also full β-reduction and η-reduction. Moreover, as we can perform calculations between several substitutions being evaluated, C.C.L. appears more powerful than the Lambda-Calculus. All the strategies of the Lambda-Calculus may be straightforwardly translated into derivations of C.C.L. We are currently studying other strategies using the new capabilities for substitution. A first approch of this problem may be found in [9].

Related works

H.Yokouchi [20] develops another approach. He deals with λ-calculus with variables. His translation from λ-calculus to C.C.L. is essentially the one defined by $()_{DB_{x_1,\ldots,x_n}}$. His translation from C.C.L. to λ-calculus is completely different from our translation $()_\lambda$. A C.C.L. term F is seen as a function, say M. Let N be a λ-term. To F and N, is associated a λ-term $F^\star[N]$ which is intended to represent $M[x \longleftarrow N]$ and which is the translation of F in λ-calculus. Now, if $F \xrightarrow{SL\beta} G$, then $F^\star[N] \xrightarrow{\beta} G^\star[N]$. Any term H of C.C.L. such that $H^\star_{DB(x,y)} \equiv H^\star_{DB(x)}$ and such that any sub-term App appears in the pattern $App \circ <\,,\,>$ is said to be regular. This set is different from

$\mathcal{D}$. $SL\beta$ is shown confluent on the set of regular terms by using the Church-Rosser Property for λ-calculus.

However, these translations do not define a bijection between $\Lambda(V)$ and a subset of C.C.L. and between β-derivations and a subset of $SL\beta$-derivations.

Contents

References

[1] H. P. Barendregt. *The Lambda-Calculus.* Volume 103, Elsevier Science Publishing Company, Amsterdam, 1984.

[2] H. P. Barendregt. *Pairing without conventional restraints.* Zeitschr. J. Math And Logik und Grundlagen p Math, 20:289–306, 1974.

[3] N. de Bruijn. *Lambda-Calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser Theorem.* Indag. Math., 34(5):381–392, 1972.

[4] N. de Bruijn. *Lambda-Calculus notation with namefree formulas involing symbols that represent reference transforming mappings* Indag. Math.,40 :348–356, 1978.

[5] A. Church. *The Calculi of Lambda-Conversion.* Ann. of Math. Studies, 6, 1941.

[6] P.L. Curien. *Categorical Combinators, Sequential Algorithms and Functional Programming.* Research Notes in Theoretical Computer Science, Pitman, London, 1986.

[7] H. B. Curry. *Combinatory Logic.* Volume 1, North-Holland, 1958.

[8] H. B. Curry, J. R. Hindley, and J. P. Seldin. *Combinatory Logic.* Volume 2, North-Holland, 1972.

[9] T.Hardin. *Résultats de confluence pour les Règles fortes de la Logique Combinatoire Catégorique et Liens avec les Lambda-calculs* Thèse de Doctorat, Université de Paris 7, 1987

[10] T. Hardin et A. Laville. *Proof of Termination of The Rewriting System Subst on C.C.L.* T.C.S., 46:305–312, 1986.

[11] R. Hindley and J. Seldin. *Introduction to Combinators and λ-calculus.* Volume 1 of *London Mathematical Society Student texts*, Cambridge University Press, 1986.

[12] J. W. Klop. *Combinatory Reduction Systems.* Dissertation, Mathematisch Centrum Amsterdam, 1982.

[13] J. W. Klop.,R. de. Vrijer. *Unique Normal Forms for λ-calculus with Surjective Pairing* ,Rapport 87-03. Centre of Mathematics and Computer Science, Amsterdam, 1987.

[14] D. Knuth and P. Bendix. *Simple Word Problems in Universal Algebras.* In J. Leech, editor, Computational Problems in Abstract Algebra, pages 263–297, Pergamon, 1970.

[15] J. Lambek, P.J. Scott *Introduction to Higher Order Categorical Logic* ,Cambridge University Press,Cambridge Amsterdam, 1987.

[16] J. J. Levy. *Réductions correctes and optimales dans le Lambda-Calcul.* Thèse d'Etat, Université de Paris 7, 1978.

[17] M. Mauny. *Compilation des Langages Fonctionnels dans les Combinateurs Categoriques.* Thèse de Troisième Cycle, Université Paris 7, Septembre 1985.

[18] D. Turner. *A new implementation technique for applicative languages.* Software Practice and Experience, 9:31–49, 1979.

[19] R. de Vrijer. *Surjective Pairing and Strong Normalisation:two themes in λ-calculus* Dissertation, University of Amsterdam.

[20] H. Yokoucki. *Relationship between λ-calculus and Rewriting Systems for Categorical Combinators* Preprint, I.B.M. Research Center. Tokyo.

Imprimé en France
par
l'Institut National de Recherche en Informatique et en Automatique